Jingling Xue

List of Publications by Year
in descending order

Source: https://exaly.com/author-pdf/7362703/publications.pdf

Version: 2024-02-01

240

papers

3,431

citations

331670

21

h-index

395702

33

g-index

245

all docs

245

docs citations

245

times ranked

1306

citing authors

#	Article	IF	Citations
1	SVF: interprocedural static value-flow analysis in LLVM. , 2016, , .		203
2	Loop Tiling for Parallelism. , 2000, , .		137
3	Static memory leak detection using full-sparse value-flow analysis. , 2012, , .		94
4	Data cache locking for higher program predictability. , 2003, , .		80
5	Compiler-directed scratchpad memory management via graph coloring. Transactions on Architecture and Code Optimization, 2009, 6, 1-17.	2.0	79
6	Level by level. , 2010, , .		74
7	Detecting Memory Leaks Statically with Full-Sparse Value-Flow Analysis. IEEE Transactions on Software Engineering, 2014, 40, 107-122.	5.6	69
8	The Reliability Wall for Exascale Supercomputing. IEEE Transactions on Computers, 2012, 61, 767-779.	3.4	67
9	On Tiling as a Loop Transformation. Parallel Processing Letters, 1997, 07, 409-424.	0.6	65
10	On-demand dynamic summary-based points-to analysis. , 2012, , .		63
11	Memory coloring: a compiler approach for scratchpad memory management. , 2005, , .		59
12	Communication-Minimal Tiling of Uniform Dependence Loops. Journal of Parallel and Distributed Computing, 1997, 42, 42-59.	4.1	55
13	Comparison of quantum dots immunofluorescence histochemistry and conventional immunohistochemistry for the detection of caveolin-1 and PCNA in the lung cancer tissue microarray. Journal of Molecular Histology, 2009, 40, 261-268.	2.2	55
14	On-demand strong update analysis via value-flow refinement. , 2016, , .		55
15	Understanding and detecting evolution-induced compatibility issues in Android apps. , 2018, , .		52
16	An Incremental Points-to Analysis with CFL-Reachability. Lecture Notes in Computer Science, 2013, , 61-81.	1.3	50
17	Data cache locking for tight timing calculations. Transactions on Embedded Computing Systems, 2007, 7, 1-38.	2.9	49
18	Efficient and precise points-to analysis: modeling the heap by merging equivalent automata. , 2017, , .		48

#	Article	IF	Citations
19	Making k-Object-Sensitive Pointer Analysis More Precise with Still k-Limiting. Lecture Notes in Computer Science, 2016, , 489-510.	1.3	43
20	Spatio-temporal context reduction. , 2018, , .		42
21	Automating non-unimodular loop transformations for massive parallelism. Parallel Computing, 1994, 20, 711-728.	2.1	39
22	Let's study whole-program cache behaviour analytically. , 0, , .		38
23	Durable Address Translation in PCM-based Flash Storage Systems. IEEE Transactions on Parallel and Distributed Systems, 2016, , 1-1.	5.6	37
24	Sparse flow-sensitive pointer analysis for multithreaded programs. , 2016, , .		37
25	Data cache locking for higher program predictability. Performance Evaluation Review, 2003, 31, 272-282.	0.6	36
26	A Compiler Approach for Exploiting Partial SIMD Parallelism. Transactions on Architecture and Code Optimization, 2016, 13, 1-26.	2.0	33
27	Self-inferencing Reflection Resolution for Java. Lecture Notes in Computer Science, 2014, , 27-53.	1.3	33
28	Exploiting mixed SIMD parallelism by reducing data reorganization overhead. , 2016, , .		32
29	Acculock: Accurate and efficient detection of data races. , 2011, , .		29
30	Enhanced Peripheral Nerve Regeneration by a High Surface Area to Volume Ratio of Nerve Conduits Fabricated from Hydroxyethyl Cellulose/Soy Protein Composite Sponges. ACS Omega, 2017, 2, 7471-7481.	3.5	29
31	Reuse-Driven Tiling for Improving Data Locality. International Journal of Parallel Programming, 1998, 26, 671-696.	1.5	28
32	SPAS: Scalable Path-Sensitive Pointer Analysis on Full-Sparse SSA. Lecture Notes in Computer Science, 2011, , 155-171.	1.3	28
33	Data caches in multitasking hard real-time systems. , 0, , .		27
34	Understanding and Analyzing Java Reflection. ACM Transactions on Software Engineering and Methodology, 2019, 28, 1-50.	6.0	27
35	Precision-preserving yet fast object-sensitive pointer analysis with partial context sensitivity. , 2019, 3, 1-29.		27
36	VFix: Value-Flow-Guided Precise Program Repair for Null Pointer Dereferences. , 2019, , .		26

#	Article	IF	Citations
37	Machine-Learning-Guided Typestate Analysis for Static Use-After-Free Detection. , 2017, , .		25
38	Automatic Library Generation for BLAS3 on GPUs. , 2011, , .		24
39	SEED: A Statically Greedy and Dynamically Adaptive Approach for Speculative Loop Execution. IEEE Transactions on Computers, 2013, 62, 1004-1016.	3.4	24
40	Boosting the precision of virtual call integrity protection with partial pointer analysis for C++. , 2017, , .		24
41	A Heterogeneous PIM Hardware-Software Co-Design for Energy-Efficient Graph Processing. , 2020, , .		24
42	Efficient and accurate analytical modeling of whole-program data cache behavior. IEEE Transactions on Computers, 2004, 53, 547-566_3.	3.4	23
43	Spara: An Energy-Efficient ReRAM-Based Accelerator for Sparse Graph Analytics Applications. , 2020, , .		22
44	Fusion-Catalyzed Pruning for Optimizing Deep Learning on Intelligent Edge Devices. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2020, 39, 3614-3626.	2.7	22
45	A Locality-Aware Energy-Efficient Accelerator for Graph Mining Applications. , 2020, , .		22
46	Parallel Pointer Analysis with CFL-Reachability. , 2014, , .		21
47	Value-Flow-Based Demand-Driven Pointer Analysis for C and C++. IEEE Transactions on Software Engineering, 2020, 46, 812-835.	5.6	21
48	DNNTune. Transactions on Architecture and Code Optimization, 2019, 16, 1-26.	2.0	21
49	The synthesis of control signals for one-dimensional systolic arrays. The Integration VLSI Journal, 1992, 14, 1-32.	2.1	19
50	Unimodular transformations of non-perfectly nested loops. Parallel Computing, 1997, 22, 1621-1645.	2.1	19
51	Effective Soundness-Guided Reflection Analysis. Lecture Notes in Computer Science, 2015, , 162-180.	1.3	19
52	Optimal loop parallelization for maximizing iteration-level parallelism. , 2009, , .		18
53	Fast and precise points-to analysis with incremental CFL-reachability summarisation: preliminary experience. , 2012, , .		18
54	Parallelizing SOR for GPGPUs using alternate loop tiling. Parallel Computing, 2012, 38, 310-328.	2.1	18

#	Article	IF	Citations
55	<scp>Acculock</scp>: accurate and efficient detection of data races. Software - Practice and Experience, 2013, 43, 543-576.	3.6	18
56	Query-directed adaptive heap cloning for optimizing compilers. , 2013, , .		18
57	Performance-Boosting Sparsification of the IFDS Algorithm with Applications to Taint Analysis. , 2019, , .		18
58	Software-Hardware Cooperative DRAM Bank Partitioning for Chip Multiprocessors. Lecture Notes in Computer Science, 2010, , 329-343.	1.3	18
59	Scratchpad allocation for data aggregates in superperfect graphs. , 2007, , .		18
60	A lifetime optimal algorithm for speculative PRE. Transactions on Architecture and Code Optimization, 2006, 3, 115-155.	2.0	17
61	A Highly Parallel Reuse Distance Analysis Algorithm on GPUs. , 2012, , .		17
62	Accelerating Dynamic Detection of Uses of Undefined Values with Static Value-Flow Analysis. , 2014, , .		17
63	Region-Based Selective Flow-Sensitive Pointer Analysis. Lecture Notes in Computer Science, 2014, , 319-336.	1.3	17
64	Optimizing deep neural networks on intelligent edge accelerators via flexible-rate filter pruning. Journal of Systems Architecture, 2022, 124, 102431.	4.3	17
65	Minimal placement of bank selection instructions for partitioned memory architectures. Transactions on Embedded Computing Systems, 2008, 7, 1-32.	2.9	16
66	Optimal WCET-aware code selection for scratchpad memory. , 2010, , .		16
67	WCET-aware data selection and allocation for scratchpad memory. , 2012, , .		16
68	Programming for scientific computing on peta-scale heterogeneous parallel systems. Journal of Central South University, 2013, 20, 1189-1203.	3.0	16
69	WPBOUND: Enforcing Spatial Memory Safety Efficiently at Runtime with Weakest Preconditions. , 2014, , .		16
70	Launch-mode-aware context-sensitive activity transition analysis. , 2018, , .		16
71	Minimizing bank selection instructions for partitioned memory architecture. , 2006, , .		16
72	Predicting Cross-Core Performance Interference on Multicore Processors with Regression Analysis. IEEE Transactions on Parallel and Distributed Systems, 2016, 27, 1443-1456.	5.6	15

#	Article	IF	Citations
73	Time-minimal tiling when rise is larger than zero. Parallel Computing, 2002, 28, 915-939.	2.1	14
74	Detecting memory errors at runtime with source-level instrumentation. , 2019, , .		14
75	Scratchpad memory allocation for data aggregates via interval coloring in superperfect graphs. Transactions on Embedded Computing Systems, 2010, 10, 1-42.	2.9	13
76	Optimally Maximizing Iteration-Level Loop Parallelism. IEEE Transactions on Parallel and Distributed Systems, 2012, 23, 564-572.	5.6	13
77	Making context-sensitive inclusion-based pointer analysis practical for compilers using parameterised summarisation. Software - Practice and Experience, 2014, 44, 1485-1510.	3.6	13
78	OpenMC: Towards Simplifying Programming for TianHe Supercomputers. Journal of Computer Science and Technology, 2014, 29, 532-546.	1.5	13
79	Reuse-driven tiling for data locality. Lecture Notes in Computer Science, 1998, , 16-33.	1.3	12
80	Partitioning and scheduling loops on NOWs. Computer Communications, 1999, 22, 1017-1033.	5.1	12
81	Automatic Parallelization of Tiled Loop Nests with Enhanced Fine-Grained Parallelism on GPUs. , 2012, , .		12
82	Eliminating Redundant Bounds Checks in Dynamic Buffer Overflow Detection Using Weakest Preconditions. IEEE Transactions on Reliability, 2016, 65, 1682-1699.	4.6	12
83	Fine grained, direct access file system support for storage class memory. Journal of Systems Architecture, 2017, 72, 80-92.	4.3	12
84	Validity Invariants and Effects. Lecture Notes in Computer Science, 2007, , 202-226.	1.3	12
85	Loop-oriented array- and field-sensitive pointer analysis for automatic SIMD vectorization. , 2016, , .		12
86	Generating efficient tiled code for distributed memory machines. Parallel Computing, 2000, 26, 1369-1410.	2.1	11
87	What Is System Hang and How to Handle It. , 2012, , .		11
88	Design and Implementation of a Highly Efficient DGEMM for 64-Bit ARMv8 Multi-core Processors. , 2015, , .		11
89	Automated memory leak fixing on value-flow slices for C programs. , 2016, , .		11
90	Model-Driven Tile Size Selection for DOACROSS Loops on GPUs. Lecture Notes in Computer Science, 2011, , 401-412.	1.3	11

#	Article	IF	Citations
91	Transformations of nested loops with non-convex iteration spaces. Parallel Computing, 1996, 22, 339-368.	2.1	10
92	Toward Harnessing DOACROSS Parallelism for Multi-GPGPUs. , 2010, , .		10
93	Extendable pattern-oriented optimization directives. Transactions on Architecture and Code Optimization, 2012, 9, 1-37.	2.0	10
94	Hadoop+. , 2015, , .		10
95	Ripple. , 2017, , .		10
96	GoBench: A Benchmark Suite of Real-World Go Concurrency Bugs. , 2021, , .		10
97	Runtime detection of memory errors with smart status. , 2021, , .		10
98	Extendable pattern-oriented optimization directives. , 2011, , .		9
99	A Fast Parallel Implementation of Molecular Dynamics with the Morse Potential on a Heterogeneous Petascale Supercomputer. , 2012, , .		9
100	Layout-oblivious compiler optimization for matrix computations. Transactions on Architecture and Code Optimization, 2013, 9, 1-20.	2.0	9
101	Accelerating inclusion-based pointer analysis on heterogeneous CPU-GPU systems. , 2013, , .		9
102	Accelerating Dynamic Detection of Uses of Undefined Values with Static Value-Flow Analysis. , 2014, , .		9
103	A collaborative divide-and-conquer K-means clustering algorithm for processing large data. , 2014, , .		9
104	Ripple : Reflection analysis for Android apps in incomplete information environments. Software - Practice and Experience, 2018, 48, 1419-1437.	3.6	9
105	May-happen-in-parallel analysis with static vector clocks. , 2018, , .		9
106	A Fresh Look at PRE as a Maximum Flow Problem. Lecture Notes in Computer Science, 2006, , 139-154.	1.3	9
107	An efficient heuristic for instruction scheduling on clustered vliw processors. , 2011, , .		8
108	A Hybrid Circular Queue Method for Iterative Stencil Computations on GPUs. Journal of Computer Science and Technology, 2012, 27, 57-74.	1.5	8

#	Article	IF	Citations
109	Region-Based May-Happen-in-Parallel Analysis for C Programs. , 2015, , .		8
110	An Efficient GPU Implementation of Inclusion-Based Pointer Analysis. IEEE Transactions on Parallel and Distributed Systems, 2016, 27, 353-366.	5.6	8
111	Reflection Analysis for Java: Uncovering More Reflective Targets Precisely. , 2017, , .		8
112	TDroid: exposing app switching attacks in Android with control flow specialization. , 2018, , .		8
113	Precise Static Happens-Before Analysis for Detecting UAF Order Violations in Android. , 2019, , .		8
114	A Feature-Oriented Corpus for Understanding, Evaluating and Improving Fuzz Testing. , 2019, , .		8
115	Eagle. ACM Transactions on Software Engineering and Methodology, 2021, 30, 1-46.	6.0	8
116	Optimizing scientific application loops on stream processors. , 2008, , .		8
117	Efficient and precise points-to analysis: modeling the heap by merging equivalent automata. ACM SIGPLAN Notices, 2017, 52, 278-291.	0.2	8
118	Comparability graph coloring for optimizing utilization of stream register files in stream processors. , 2009, , .		7
119	Exploiting loop-dependent stream reuse for stream processors. , 2008, , .		7
120	RegTT: Accelerating Tree Traversals on GPUs by Exploiting Regularities. , 2016, , .		7
121	Dynamic symbolic execution for polymorphism. , 2017, , .		7
122	Completeness Analysis for Incomplete Object-Oriented Programs. Lecture Notes in Computer Science, 2005, , 271-286.	1.3	7
123	Efficient Energy Balancing Aware Multiple Base Station Deployment for WSNs. Lecture Notes in Computer Science, 2011, , 179-194.	1.3	7
124	Enabling Loop Fusion and Tiling for Cache Performance by Fixing Fusion-Preventing Data Dependences. , 0, , .		6
125	Comparability graph coloring for optimizing utilization of stream register files in stream processors. ACM SIGPLAN Notices, 2009, 44, 111-120.	0.2	6
126	Comparability Graph Coloring for Optimizing Utilization of Software-Managed Stream Register Files for Stream Processors. Transactions on Architecture and Code Optimization, 2012, 9, 1-30.	2.0	6

#	Article	IF	Citations
127	PartialRC: A Partial Recomputing Method for Efficient Fault Recovery on GPGPUs. Journal of Computer Science and Technology, 2012, 27, 240-255.	1.5	6
128	Epipe: A low-cost fault-tolerance technique considering WCET constraints. Journal of Systems Architecture, 2013, 59, 1383-1393.	4.3	6
129	Reducing Static Energy in Supercomputer Interconnection Networks Using Topology-Aware Partitioning. IEEE Transactions on Computers, 2016, 65, 2588-2602.	3.4	6
130	An Efficient WCET-Aware Instruction Scheduling and Register Allocation Approach for Clustered VLIW Processors. Transactions on Embedded Computing Systems, 2017, 16, 1-21.	2.9	6
131	A Conflict-free Scheduler for High-performance Graph Processing on Multi-pipeline FPGAs. Transactions on Architecture and Code Optimization, 2020, 17, 1-26.	2.0	6
132	Detecting TensorFlow Program Bugs in Real-World Industrial Environment. , 2021, , .		6
133	Context Debloating for Object-Sensitive Pointer Analysis. , 2021, , .		6
134	ScalaGraph: A Scalable Accelerator for Massively Parallel Graph Processing. , 2022, , .		6
135	SPECIFYING CONTROL SIGNALS FOR SYSTOLIC ARRAYS BY UNIFORM RECURRENCE EQUATIONS. Parallel Processing Letters, 1991, 01, 83-93.	0.6	5
136	Interprocedural side-effect analysis for incomplete object-oriented software modules. Journal of Systems and Software, 2007, 80, 92-105.	4.5	5
137	Factorization of singular integer matrices. Linear Algebra and Its Applications, 2008, 428, 1046-1055.	0.9	5
138	A Cache-Efficient Parallel Gauss-Seidel Solver with Alternating Tiling. , 2009, , .		5
139	Enhancement of cooperation between file systems and applications â€" on VFS extensions for optimized performance. Science China Information Sciences, 2015, 58, 1-10.	4.3	5
140	Automatic generation of fast BLAS3-GEMM: A portable compiler approach. , 2017, , .		5
141	Revisiting Loop Tiling for Datacenters. , 2018, , .		5
142	Event trace reduction for effective bug replay of Android apps via differential GUI state analysis. , 2019, , .		5
143	SCP. Transactions on Architecture and Code Optimization, 2018, 15, 1-21.	2.0	5
144	CloudRaid: Detecting Distributed Concurrency Bugs via Log Mining and Enhancement. IEEE Transactions on Software Engineering, 2022, 48, 662-677.	5.6	5

#	Article	IF	Citations
145	Unleashing the Low-Precision Computation Potential of Tensor Cores on GPUs. , 2021, , .		5
146	Towards Data Tiling for Whole Programs in Scratchpad Memory Allocation. Lecture Notes in Computer Science, 2007, , 63-74.	1.3	5
147	Exploiting Speculative TLP in Recursive Programs by Dynamic Thread Prediction. Lecture Notes in Computer Science, 2009, , 78-93.	1.3	5
148	Bandwidth-Aware Loop Tiling for DMA-Supported Scratchpad Memory. , 2020, , .		5
149	Efficient compile-time analysis of cache behaviour for programs with IF statements. , 0, , .		4
150	Code tiling for improving the cache performance of PDE solvers. , 2003, , .		4
151	Scratchpad allocation for data aggregates in superperfect graphs. ACM SIGPLAN Notices, 2007, 42, 207-216.	0.2	4
152	Thread-Sensitive Modulo Scheduling for Multicore Processors. , 2008, , .		4
153	WCET-aware data selection and allocation for scratchpad memory. ACM SIGPLAN Notices, 2012, 47, 41-50.	0.2	4
154	Acyclic orientation graph coloring for software-managed memory allocation. Science China Information Sciences, 2014, 57, 1-18.	4.3	4
155	Loop-oriented array- and field-sensitive pointer analysis for automatic SIMD vectorization. ACM SIGPLAN Notices, 2016, 51, 41-51.	0.2	4
156	Parallel construction of interprocedural memory SSA form. Journal of Systems and Software, 2018, 146, 186-195.	4.5	4
157	TCD: Statically Detecting Type Confusion Errors in C++ Programs. , 2019, , .		4
158	A trace-based binary compilation framework for energy-aware computing. , 2004, , .		4
159	Selective Context-Sensitivity for k-CFA with CFL-Reachability. Lecture Notes in Computer Science, 2021, , 261-285.	1.3	4
160	Ownership Downgrading for Ownership Types. Lecture Notes in Computer Science, 2009, , 144-160.	1.3	4
161	Burn after reading. , 2020, , .		4
162	Buddy Stacks: Protecting Return Addresses with Efficient Thread-Local Storage and Runtime Re-Randomization. ACM Transactions on Software Engineering and Methodology, 2022, 31, 1-37.	6.0	4

#	Article	IF	Citations
163	Closed-form mapping conditions for the synthesis of linear processor arrays. Journal of Signal Processing Systems, 1995, 10, 181-199.	1.0	3
164	Optimal and efficient speculation-based partial redundancy elimination. , 0, , .		3
165	Improving the parallelism of iterative methods byÂaggressive loop fusion. Journal of Supercomputing, 2008, 43, 147-164.	3.6	3
166	Improving scratchpad allocation with demand-driven data tiling. , 2010, , .		3
167	Exploiting the reuse supplied by loop-dependent stream references for stream processors. Transactions on Architecture and Code Optimization, 2010, 7, 1-35.	2.0	3
168	Contention-Aware Scheduling for Asymmetric Multicore Processors. , 2015, , .		3
169	An Energy-Efficient Implementation of LU Factorization on Heterogeneous Systems. , 2016, , .		3
170	Loop-Oriented Pointer Analysis for Automatic SIMD Vectorization. Transactions on Embedded Computing Systems, 2018, 17, 1-31.	2.9	3
171	Towards concurrency race debugging. , 2018, , .		3
172	LCCFS: a lightweight distributed file system for cloud computing without journaling and metadata services. Science China Information Sciences, 2019, 62, 1.	4.3	3
173	Aggressive Loop Fusion for Improving Locality and Parallelism. Lecture Notes in Computer Science, 2005, , 224-238.	1.3	3
174	Ownership Types for Object Synchronisation. Lecture Notes in Computer Science, 2012, , 18-33.	1.3	3
175	A Type and Effect System for Determinism in Multithreaded Programs. Lecture Notes in Computer Science, 2012, , 518-538.	1.3	3
176	Correlating UI Contexts with Sensitive API Calls: Dynamic Semantic Extraction and Analysis. , 2020, , .		3
177	Selecting Context-Sensitivity Modularly for Accelerating Object-Sensitive Pointer Analysis. IEEE Transactions on Software Engineering, 2023, 49, 719-742.	5.6	3
178	A systolic array for pyramidal algorithms. Journal of Signal Processing Systems, 1991, 3, 237-257.	1.0	2
179	CoopStream: A Cooperative Cache Based Streaming Schedule Scheme for On-demand Media Services on Overlay Networks. , 0, , .		2
180	Toward Automatic Data Distribution for Migrating Computations. Parallel Processing (ICPP), Proceedings of the International Symposium, 2007, , .	0.0	2

#	Article	IF	Citations
181	Loop recreation for thread-level speculation. , 2007, , .		2
182	Optimizing scientific application loops on stream processors. ACM SIGPLAN Notices, 2008, 43, 161-170.	0.2	2
183	Hardware Support for Efficient Sparse Matrix Vector Multiplication. , 2008, , .		2
184	Loop recreation for threadâ€level speculation on multicore processors. Software - Practice and Experience, 2010, 40, 45-72.	3.6	2
185	Reuse-aware modulo scheduling for stream processors. , 2010, , .		2
186	The case for a scalable coherence protocol for complex on-chip cache hierarchies in many core systems. , 2013, , .		2
187	Performance Modeling of Multithreaded Programs for Mobile Asymmetric Chip Multiprocessors. , 2015, , .		2
188	Live path control flow integrity. , 2018, , .		2
189	Improving the Performance of GCC by Exploiting IA-64 Architectural Features. Lecture Notes in Computer Science, 2005, , 236-251.	1.3	2
190	Structural Lock Correlation with Ownership Types. Lecture Notes in Computer Science, 2013, , 391-410.	1.3	2
191	Region-Based Partial Dead Code Elimination on Predicated Code. Lecture Notes in Computer Science, 2004, , 150-166.	1.3	2
192	Every Mutation Should Be Rewarded: Boosting Fault Localization with Mutated Predicates. , 2020, , .		2
193	Exposing Android Event-Based Races by Selective Branch Instrumentation. , 2020, , .		2
194	Loop2Recursion: Compiler-Assisted Wear Leveling for Non-Volatile Memory. , 2020, , .		2
195	Eigenvectors-based parallelisation of nested loops with affine dependences. , 0, , .		1
196	Space-Time Equations for Non-Unimodular Mappings. International Journal of Computer Mathematics, 2002, 79, 555-572.	1.8	1
197	Instruction Scheduling with Release Times and Deadlines on ILP Processors. , 2006, , .		1
198	Partial dead code elimination on predicated code regions. Software - Practice and Experience, 2006, 36, 1655-1685.	3.6	1

#	Article	IF	Citations
199	Trace-based leakage energy optimisations at link time. Journal of Systems Architecture, 2007, 53, 1-20.	4.3	1
200	ACS: An Addressless Configuration Support for efficient partial reconfigurations. , 2008, , .		1
201	A gather/scatter hardware support for efficient Fast Fourier Transform. , 2008, , .		1
202	PARBLO: Page-Allocation-Based DRAM Row Buffer Locality Optimization. Journal of Computer Science and Technology, 2009, 24, 1086-1097.	1.5	1
203	On Reducing Hidden Redundant Memory Accesses for DSP Applications. IEEE Transactions on Very Large Scale Integration (VLSI) Systems, 2011, 19, 997-1010.	3.1	1
204	Instruction scheduling with k-successor tree for clustered VLIW processors. Design Automation for Embedded Systems, 2013, 17, 439-458.	1.0	1
205	File system-independent block device support for storage class memory. , 2015, , .		1
206	AutoFix. ACM SIGAPP Applied Computing Review: A Publication of the Special Interest Group on Applied Computing, 2017, 16, 38-50.	0.9	1
207	Incremental Analysis for Probabilistic Programs. Lecture Notes in Computer Science, 2017, , 450-472.	1.3	1
208	Poker: permutation-based SIMD execution of intensive tree search by path encoding. , 2018, , .		1
209	Live Path CFI Against Control Flow Hijacking Attacks. Lecture Notes in Computer Science, 2018, , 768-779.	1.3	1
210	Incremental precision-preserving symbolic inference for probabilistic programs. , 2019, , .		1
211	WCET-aware hyper-block construction for clustered VLIW processors. , 2019, , .		1
212	PPOpenCL: a performance-portable OpenCL compiler with host and kernel thread code fusion. , 2019, , .		1
213	On the loading, recovery and access of stationary data in systolic arrays. Lecture Notes in Computer Science, 1992, , 259-264.	1.3	1
214	Communication-minimal tiling of uniform dependence loops. Lecture Notes in Computer Science, 1997, , 330-349.	1.3	1
215	Automatic Synthesis of Data-Flow Analyzers. Lecture Notes in Computer Science, 2021, , 453-478.	1.3	1
216	Guest Editorial: Special Section on New Trends in Parallel and Distributed Computing for Human Sensible Applications. IEEE Transactions on Emerging Topics in Computing, 2021, 9, 1640-1641.	4.6	1

#	Article	IF	Citations
217	Recovering Container Class Types in C++ Binaries. , 2022, , .		1
218	M3V: Multi-modal Multi-view Context Embedding for Repair Operator Prediction. , 2022, , .		1
219	Non-unimodular code generation for parallel machines. , 0, , .		0
220	GENERALISING THE UNIMODULAR APPROACH TO RESTRUCTURE IMPERFECTLY NESTED LOOPS. Parallel Processing Letters, 1996, 06, 401-414.	0.6	0
221	EIGENVECTORS-BASED PARALLELISATION OF NESTED LOOPS WITH AFFINE DEPENDENCES. International Journal of Parallel, Emergent and Distributed Systems, 2002, 17, 227-248.	0.4	0
222	Code Tiling: One Size Fits All. , 2006, , 219-240.		0
223	Advances in high performance computing. Journal of Supercomputing, 2008, 43, 105-106.	3.6	0
224	Gather/scatter hardware support for accelerating Fast Fourier Transform. Journal of Systems Architecture, 2010, 56, 667-684.	4.3	0
225	Leakage-Aware Modulo Scheduling for Embedded VLIW Processors. Journal of Computer Science and Technology, 2011, 26, 405-417.	1.5	0
226	Layout-oblivious optimization for matrix computations. , 2012, , .		0
227	Optimizing modulo scheduling to achieve reuse andÂconcurrency for stream processors. Journal of Supercomputing, 2012, 59, 1229-1251.	3.6	0
228	Scratchpad Memory aware task scheduling with minimum number of preemptions on a single processor. , 2013, , .		0
229	Lifetime holes aware register allocation for clustered VLIW processors. , 2014, , .		0
230	Masking Soft Errors with Static Bitwise Analysis. , 2016, , .		0
231	May-happen-in-parallel analysis with static vector clocks. , 2018, , .		0
232	P oker. Transactions on Architecture and Code Optimization, 2019, 15, 1-28.	2.0	0
233	Referee: A Pattern-Guided Approach for Auto Design in Compiler-Based Analyzers. , 2020, , .		0
234	TIME-MINIMAL AND PROCESSOR-TIME-MINIMAL LOOP TILING. , 2000, , .		0

#	Article	IF	Citations
235	A trace-based binary compilation framework for energy-aware computing. ACM SIGPLAN Notices, 2004, 39, 95-106.	0.2	0
236	Lifetime holes aware register allocation for clustered VLIW processors. , 2014, , .		0
237	Poker: permutation-based SIMD execution of intensive tree search by path encoding. , 2018, , .		0
238	Per-Dereference Verification of Temporal Heap Safety via Adaptive Context-Sensitive Analysis. Lecture Notes in Computer Science, 2019, , 48-72.	1.3	0
239	VTensor. , 2020, , .		0
240	A Dynamic Analysis Tool for Memory Safety Based on Smart Status and Source-Level Instrumentation. , 2022, , .		0