Mark Harman

List of Publications by Year
in descending order

Source: https://exaly.com/author-pdf/5202732/publications.pdf

Version: 2024-02-01

289	17,840	53794 45	30087 103
papers	citations	h-index	g-index

300	300	300	4968
all docs	docs citations	times ranked	citing authors

#	Article	IF	Citations
1	Learning From Mistakes: Machine Learning Enhanced Human Expert Effort Estimates. IEEE Transactions on Software Engineering, 2022, 48, 1868-1882.	5.6	6
2	Machine Learning Testing: Survey, Landscapes and Horizons. IEEE Transactions on Software Engineering, 2022, 48, 1-36.	5.6	315
3	A Survey of Performance Optimization for Mobile Applications. IEEE Transactions on Software Engineering, 2022, 48, 2879-2904.	5.6	21
4	FAUSTA: Scaling Dynamic Analysis with Traffic Generation at WhatsApp. , 2022, , .		3
5	Improving machine translation systems via isotopic replacement. , 2022, , .		16
6	Comparative Analysis of Constraint Handling Techniques for Constrained Combinatorial Testing. IEEE Transactions on Software Engineering, 2021, 47, 2549-2562.	5.6	10
7	An Empirical Validation of Oracle Improvement. IEEE Transactions on Software Engineering, 2021, 47, 1708-1728.	5.6	8
8	A Study of Bug Resolution Characteristics in Popular Programming Languages. IEEE Transactions on Software Engineering, 2021, 47, 2684-2697.	5.6	10
9	App Store Effects on Software Engineering Practices. IEEE Transactions on Software Engineering, 2021, 47, 300-319.	5.6	40
10	MuDelta: Delta-Oriented Mutation Testing at Commit Time. , 2021, , .		8
11	Testing Web Enabled Simulation at Scale Using Metamorphic Testing. , 2021, , .		27
12	Artifact for Enhancing Genetic Improvement of Software with Regression Test Selection. , 2021, , .		1
13	"Ignorance and Prejudice" in Software Fairness. , 2021, , .		29
14	Enhancing Genetic Improvement of Software with Regression Test Selection. , 2021, , .		11
15	Facebookâ€™s Cyberâ€“Cyber and Cyberâ€“Physical Digital Twins. , 2021, , .		17
16	Fairea: a model behaviour mutation approach to benchmarking bias mitigation methods. , 2021, , .		26
17	An Empirical Comparison of Combinatorial Testing, Random Testing and Adaptive Random Testing. IEEE Transactions on Software Engineering, 2020, 46, 302-320.	5.6	35
18	Cost measures matter for mutation testing study validity. , 2020, , .		7

#	Article	IF	Citations
19	Automatic testing and improvement of machine translation. , 2020, , .		63
20	Ownership at Large. , 2020, , .		3
21	WES. , 2020, , .		18
22	Mutation Testing Advances: An Analysis and Survey. Advances in Computers, 2019, , 275-378.	1.6	239
23	Approximate Oracles and Synergy in Software Energy Search Spaces. IEEE Transactions on Software Engineering, 2019, 45, 1150-1169.	5.6	19
24	Some challenges for software testing research (invited talk paper). , 2019, , .		5
25	The importance of accounting for real-world labelling when predicting software vulnerabilities. , 2019, , .		45
26	SapFix: Automated End-to-End Repair at Scale. , 2019, , .		85
27	Predictive Mutation Testing. IEEE Transactions on Software Engineering, 2019, 45, 898-918.	5.6	79
28	Detecting Trivial Mutant Equivalences via Compiler Optimisations. IEEE Transactions on Software Engineering, 2018, 44, 308-333.	5.6	60
29	Specialising Software for Different Downstream Applications Using Genetic Improvement and Code Transplantation. IEEE Transactions on Software Engineering, 2018, 44, 574-594.	5.6	28
30	Are mutants really natural?. , 2018, , .		9
31	OASIs: oracle assessment and improvement tool. , 2018, , .		4
32	Customer Rating Reactions Can Be Predicted Purely using App Features. , 2018, , .		31
33	Deploying Search Based Software Engineering with Sapienz at Facebook. Lecture Notes in Computer Science, 2018, , 3-45.	1.3	54
34	We Need a Testability Transformation Semantics. Lecture Notes in Computer Science, 2018, , 3-17.	1.3	10
35	An Empirical Study of Meta- and Hyper-Heuristic Search for Multi-Objective Release Planning. ACM Transactions on Software Engineering and Methodology, 2018, 27, 1-32.	6.0	26
36	Memory mutation testing. Information and Software Technology, 2017, 81, 97-111.	4.4	19

#	Article	IF	Citations
37	An experimental search-based approach to cohesion metric evaluation. Empirical Software Engineering, 2017, 22, 292-329.	3.9	15
38	Adaptive Multi-Objective Evolutionary Algorithms for Overtime Planning in Software Projects. IEEE Transactions on Software Engineering, 2017, 43, 898-917.	5.6	34
39	Inferring Automatic Test Oracles. , 2017, , .		12
40	Human Competitiveness of Genetic Programming in Spectrum-Based Fault Localisation. ACM Transactions on Software Engineering and Methodology, 2017, 26, 1-30.	6.0	59
41	The Value of Exact Analysis in Requirements Selection. IEEE Transactions on Software Engineering, 2017, 43, 580-596.	5.6	18
42	Genetic improvement of GPU software. Genetic Programming and Evolvable Machines, 2017, 18, 5-44.	2.2	28
43	A survey of the use of crowdsourcing in software engineering. Journal of Systems and Software, 2017, 126, 57-84.	4.5	243
44	Generalized observational slicing for tree-represented modelling languages. , 2017, , .		14
45	Automated search for good coverage criteria. , 2016, , .		9
46	Comparing white-box and black-box test prioritization. , 2016, , .		117
47	Evaluation of estimation models using the Minimum Interval of Equivalence. Applied Soft Computing Journal, 2016, 49, 956-967.	7.2	5
48	Threats to the validity of mutation-based test assessment. , 2016, , .		84
49	Sapienz: multi-objective automated testing for Android applications. , 2016, , .		346
50	Mutation-aware fault prediction. , 2016, , .		34
51	Test oracle assessment and improvement. , 2016, , .		34
52	An empirical study on dependence clusters for effort-aware fault-proneness prediction. , 2016, , .		23
53	API-Constrained Genetic Improvement. Lecture Notes in Computer Science, 2016, , 224-230.	1.3	10
54	HOMI: Searching Higher Order Mutants forÂSoftware Improvement. Lecture Notes in Computer Science, 2016, , 18-33.	1.3	8

#	Article	IF	Citations
55	Multi-objective software effort estimation. , 2016, , .		97
56	ORBS and the limits of static slicing. , 2015, , .		25
57	Introduction to the special issue on Mutation Testing. Software Testing Verification and Reliability, 2015, 25, 461-463.	2.0	2
58	An Integer Linear Programming approach to the single and bi-objective Next Release Problem. Information and Software Technology, 2015, 65, 1-13.	4.4	53
59	Feature lifecycles as they spread, migrate, remain, and die in App Stores. , 2015, , .		41
60	Trivial Compiler Equivalence: A Large Scale Empirical Study of a Simple, Fast and Effective Equivalent Mutant Detection Technique. , 2015, , .		81
61	Mutation testing of memory-related operators. , 2015, , .		8
62	Transformed Vargha-Delaney Effect Size. Lecture Notes in Computer Science, 2015, , 318-324.	1.3	24
63	Multi-objective Module Clustering for Kate. Lecture Notes in Computer Science, 2015, , 282-288.	1.3	6
64	The App Sampling Problem for App Store Mining. , 2015, , .		62
65	Learning Combinatorial Interaction Test Generation Strategies Using Hyperheuristic Search. , 2015, , .		41
66	Optimizing Existing Software With Genetic Programming. IEEE Transactions on Evolutionary Computation, 2015, 19, 118-135.	10.0	147
67	The Oracle Problem in Software Testing: A Survey. IEEE Transactions on Software Engineering, 2015, 41, 507-525.	5.6	608
68	Achievements, Open Problems and Challenges for Search Based Software Testing. , 2015, , .		124
69	GI4GI. , 2015, , .		5
70	Practical Combinatorial Interaction Testing: Empirical Findings on Efficiency and Early Fault Detection. IEEE Transactions on Software Engineering, 2015, 41, 901-924.	5.6	80
71	Combining Multi-Objective Search and Constraint Solving for Configuring Large Software Product Lines. , 2015, , .		62
72	Deep Parameter Optimisation. , 2015, , .		74

#	Article	IF	Citations
73	Reducing Energy Consumption Using Genetic Improvement. , 2015, , .		80
74	Empirical evaluation of pareto efficient multi-objective regression test case prioritisation. , 2015, , .		61
75	Improving CUDA DNA Analysis Software with Genetic Programming. , 2015, , .		46
76	Automated software transplantation. , 2015, , .		90
77	App store mining and analysis. , 2015, , .		13
78	Genetic Improvement using Higher Order Mutation. , 2015, , .		8
79	Grow and Graft a Better CUDA pknotsRG for RNA Pseudoknot Free Energy Calculation. , 2015, , .		24
80	Inferring Test Models from Kateâ€™s Bug Reports Using Multi-objective Search. Lecture Notes in Computer Science, 2015, , 301-307.	1.3	10
81	Regression Test Case Prioritisation for Guava. Lecture Notes in Computer Science, 2015, , 221-227.	1.3	5
82	Automated Transplantation of Call Graph and Layout Features into Kate. Lecture Notes in Computer Science, 2015, , 262-268.	1.3	17
83	Genetic improvement for adaptive software engineering (keynote). , 2014, , .		35
84	Robust next release problem. , 2014, , .		25
85	The plastic surgery hypothesis. , 2014, , .		136
86	Exact scalable sensitivity analysis for the next release problem. ACM Transactions on Software Engineering and Methodology, 2014, 23, 1-31.	6.0	29
87	Coverage and fault detection of the output-uniqueness test selection criteria. , 2014, , .		40
88	Improving 3D medical image registration CUDA software with genetic programming. , 2014, , .		33
89	An analysis of the relationship between conditional entropy and failed error propagation in software testing. , 2014, , .		38
90	ORBS: language-independent program slicing. , 2014, , .		57

#	Article	IF	Citations
91	Angels and monsters. , 2014, , .		30
92	A study of equivalent and stubborn mutation operators using human analysis of equivalence. , 2014, , .		110
93	Search-Based Software Project Management. , 2014, , 373-399.		26
94	Equivalence hypothesis testing in experimental software engineering. Software Quality Journal, 2014, 22, 215-238.	2.2	6
95	The executable experimental template pattern for the systematic comparison of metaheuristics. , 2014, , .		8
96	FITTEST: A new continuous and automated testing process for future Internet applications. , 2014, , .		7
97	Coherent clusters in source code. Journal of Systems and Software, 2014, 88, 1-24.	4.5	10
98	Less is More: Temporal Fault Predictive Performance over Multiple Hadoop Releases. Lecture Notes in Computer Science, 2014, , 240-246.	1.3	24
99	Babel Pidgin: SBSE Can Grow and Graft Entirely New Functionality into a Real World System. Lecture Notes in Computer Science, 2014, , 247-252.	1.3	41
100	Using Genetic Improvement and Code Transplants to Specialise a C++ Program to a Problem Class. Lecture Notes in Computer Science, 2014, , 137-149.	1.3	88
101	Genetically Improved CUDA C++ Software. Lecture Notes in Computer Science, 2014, , 87-99.	1.3	23
102	The FITTEST Tool Suite for Testing Future Internet Applications. Lecture Notes in Computer Science, 2014, , 1-31.	1.3	1
103	The FITTEST Tool Suite for Testing Future Internet Applications. Lecture Notes in Computer Science, 2014, , 1-31.	1.3	1
104	Testing and verification in serviceâ€oriented architecture: a survey. Software Testing Verification and Reliability, 2013, 23, 261-313.	2.0	87
105	GPGPU test suite minimisation: search based software engineering performance improvement using graphics cards. Empirical Software Engineering, 2013, 18, 550-593.	3.9	30
106	Foreword to the invited impact paper on automatic software repair. Software Quality Journal, 2013, 21, 419-419.	2.2	0
107	Genetic programming for Reverse Engineering. , 2013, , .		28
108	Pricing crowdsourcing-based software development tasks. , 2013, , .		37

#	Article	IF	Citations
109	Dynamic adaptive Search Based Software Engineering needs fast approximate metrics (keynote). , 2013, , .		5
110	Automated generation of state abstraction functions using data invariant inference. , 2013, , .		4
111	Amorphous Slicing of Extended Finite State Machines. IEEE Transactions on Software Engineering, 2013, 39, 892-909.	5.6	20
112	Editorial for special issue of STVR on software testing, verification, and validation - volume 2 (extended selected papers from ICST 2011). Software Testing Verification and Reliability, 2013, 23, 529-529.	2.0	0
113	Not going to take this anymore: Multi-objective overtime planning for Software Engineering projects. , 2013, , .		31
114	Empirical evaluation of search based requirements interaction management. Information and Software Technology, 2013, 55, 126-152.	4.4	48
115	AUSTIN: An open source tool for search based software testing of C programs. Information and Software Technology, 2013, 55, 112-125.	4.4	47
116	Cloud engineering is Search Based Software Engineering too. Journal of Systems and Software, 2013, 86, 2225-2241.	4.5	47
117	An orchestrated survey of methodologies for automated software test case generation. Journal of Systems and Software, 2013, 86, 1978-2001.	4.5	493
118	Searching for better configurations: a rigorous approach to clone evaluation. , 2013, , .		109
119	Empirical answers to fundamental software engineering problems (panel). , 2013, , .		3
120	Efficient Identification of Linchpin Vertices in Dependence Clusters. ACM Transactions on Programming Languages and Systems, 2013, 35, 1-35.	2.1	2
121	State-based model slicing. ACM Computing Surveys, 2013, 45, 1-36.	23.0	33
122	Efficiency and early fault detection with lower and higher strength combinatorial interaction testing. , 2013, , .		71
123	Fault localization prioritization. ACM Transactions on Software Engineering and Methodology, 2013, 22, 1-29.	6.0	83
124	Editorial for special issue of STVR on software testing, verification, and validation â€•volume 1 (extended selected papers from ICST 2011). Software Testing Verification and Reliability, 2013, 23, 437-437.	2.0	0
125	1st International workshop on combining modelling and search-based software engineering (CMSBSE) Tj ETQq1 1 0.784314 rgBT /Ov		1
126	Provably Optimal and Human-Competitive Results in SBSE for Spectrum Based Fault Localisation. Lecture Notes in Computer Science, 2013, , 224-238.	1.3	60

#	Article	IF	Citations
127	Applying Genetic Improvement to MiniSAT. Lecture Notes in Computer Science, 2013, , 257-262.	1.3	23
128	Using Genetic Algorithms to Search for Key Stakeholders in Large-Scale Software Projects. , 2013, , 118-134.		4
129	Agent-Based Modelling of Stock Markets Using Existing Order Book Data. Lecture Notes in Computer Science, 2013, , 101-114.	1.3	3
130	The GISMOE challenge: constructing the pareto program surface using genetic programming to find better programs (keynote paper). , 2012, , .		77
131	Search-based software engineering. ACM Computing Surveys, 2012, 45, 1-61.	23.0	565
132	Finding the Optimal Balance between Over and Under Approximation of Models Inferred from Execution Logs. , 2012, , .		20
133	Crawlability Metrics for Web Applications. , 2012, , .		5
134	Overview of TASE 2012 Talk on Search Based Software Engineering. , 2012, , .		0
135	Search Based Software Engineering: Techniques, Taxonomy, Tutorial. Lecture Notes in Computer Science, 2012, , 1-59.	1.3	128
136	Augmenting test suites effectiveness by increasing output diversity. , 2012, , .		22
137	Dynamic adaptive search based software engineering. , 2012, , .		58
138	App store mining and analysis: MSR for app stores. , 2012, , .		193
139	Experimental assessment of software metrics using automated refactoring. , 2012, , .		69
140	The role of Artificial Intelligence in Software Engineering. , 2012, , .		56
141	Regression testing minimization, selection and prioritization: a survey. Software Testing Verification and Reliability, 2012, 22, 67-120.	2.0	941
142	Test data regeneration: generating new test data from existing test data. Software Testing Verification and Reliability, 2012, 22, 171-201.	2.0	35
143	Regression test suite prioritization using system models. Software Testing Verification and Reliability, 2012, 22, 481-506.	2.0	30
144	Evolutionary testing of autonomous software agents. Autonomous Agents and Multi-Agent Systems, 2012, 25, 260-283.	2.1	49

#	Article	IF	Citations
145	Input Domain Reduction through Irrelevant Variable Removal and Its Effect on Local, Global, and Hybrid Search-Based Structural Test Data Generation. IEEE Transactions on Software Engineering, 2012, 38, 453-477.	5.6	45
146	Optimised Realistic Test Input Generation Using Web Services. Lecture Notes in Computer Science, 2012, , 105-120.	1.3	2
147	Refactoring as Testability Transformation. , 2011, , .		16
148	Future Internet Testing with FITTEST. , 2011, , .		5
149	Automated web application testing using search based software engineering. , 2011, , .		72
150	Automatically generating realistic test input from web services. , 2011, , .		29
151	Symbolic search-based testing. , 2011, , .		61
152	A unifying theory of control dependence and its application to arbitrary program structures. Theoretical Computer Science, 2011, 412, 6809-6842.	0.9	22
153	Software Engineering Meets Evolutionary Computation. Computer, 2011, 44, 31-39.	1.1	59
154	Software Module Clustering as a Multi-Objective Search Problem. IEEE Transactions on Software Engineering, 2011, 37, 264-282.	5.6	288
155	An Analysis and Survey of the Development of Mutation Testing. IEEE Transactions on Software Engineering, 2011, 37, 649-678.	5.6	1,135
156	A study of the bi-objective next release problem. Empirical Software Engineering, 2011, 16, 29-60.	3.9	61
157	Crawlability metrics for automated web testing. International Journal on Software Tools for Technology Transfer, 2011, 13, 131-149.	1.9	16
158	The use of searchâ€based optimization techniques to schedule and staff software projects: an approach and an empirical study. Software - Practice and Experience, 2011, 41, 495-519.	3.6	50
159	Comparing the performance of metaheuristics for the analysis of multi-stakeholder tradeoffs in requirements optimisation. Information and Software Technology, 2011, 53, 761-773.	4.4	29
160	Model projection. , 2011, , .		19
161	Transition coverage testing for simulink/stateflow models using messy genetic algorithms. , 2011, , .		11
162	Strong higher order mutation-based test data generation. , 2011, , .		104

#	Article	IF	Citations
163	Making the Case for MORTO: Multi Objective Regression Test Optimization. , 2011, , .		34
164	FlagRemover. ACM Transactions on Software Engineering and Methodology, 2011, 20, 1-33.	6.0	30
165	Cooperative Co-evolutionary Optimization of Software Project Staff Assignments and Job Scheduling. Lecture Notes in Computer Science, 2011, , 127-141.	1.3	29
166	Highly Scalable Multi Objective Test Suite Minimisation Using Graphics Cards. Lecture Notes in Computer Science, 2011, , 219-236.	1.3	44
167	An alternative characterization of weak order dependence. Information Processing Letters, 2010, 110, 939-943.	0.6	2
168	Assessing the impact of global variables on program dependence and dependence clusters. Journal of Systems and Software, 2010, 83, 96-107.	4.5	20
169	Using hybrid algorithm for Pareto efficient multi-objective test suite minimisation. Journal of Systems and Software, 2010, 83, 689-701.	4.5	115
170	An empirical investigation into branch coverage for C programs using CUTE and AUSTIN. Journal of Systems and Software, 2010, 83, 2379-2391.	4.5	50
171	Efficient multi-objective higher order mutation testing with genetic programming. Journal of Systems and Software, 2010, 83, 2416-2430.	4.5	115
172	Estimating the feasibility of transition paths inÂextended finite state machines. Automated Software Engineering, 2010, 17, 33-56.	2.9	33
173	A Theoretical and Empirical Study of Search-Based Testing: Local, Global, and Hybrid Search. IEEE Transactions on Software Engineering, 2010, 36, 226-247.	5.6	298
174	A trajectory-based strict semantics for program slicing. Theoretical Computer Science, 2010, 411, 1372-1386.	0.9	19
175	A Manifesto for Higher Order Mutation Testing. , 2010, , .		57
176	Automated patching techniques. Communications of the ACM, 2010, 53, 108-108.	4.5	24
177	The relationship between search based software engineering and predictive modeling. , 2010, , .		53
178	Coherent dependence clusters. , 2010, , .		6
179	Issues in clone classification for dataflow languages. , 2010, , .		13
180	Today/future importance analysis. , 2010, , .		17

#	Article	IF	Citations
181	Optimizing for the Number of Tests Generated in Search Based Test Data Generation with an Application to the Oracle Cost Problem. , 2010, , .		62
182	Search Based Software Engineering: Introduction to the Special Issue of the IEEE Transactions on Software Engineering. IEEE Transactions on Software Engineering, 2010, 36, 737-741.	5.6	29
183	AUSTIN: A Tool for Search Based Software Testing for the C Language and Its Evaluation on Deployed Automotive Systems. , 2010, , .		23
184	Empirical Study on the Efficiency of Search Based Test Generation for EFSM Models. , 2010, , .		19
185	Search Based Optimization of Requirements Interaction Management. , 2010, , .		24
186	Reducing qualitative human oracle costs associated with automatically generated test data. , 2010, , .		38
187	Why Source Code Analysis and Manipulation Will Always be Important. , 2010, , .		38
188	Why the Virtual Nature of Software Makes It Ideal for Search Based Optimization. Lecture Notes in Computer Science, 2010, , 1-12.	1.3	27
189	Multi objective higher order mutation testing with GP. , 2009, , .		4
190	Dependence clusters in source code. ACM Transactions on Programming Languages and Systems, 2009, 32, 1-33.	2.1	39
191	Automated test data generation for aspect-oriented programs. , 2009, , .		40
192	A search based approach to fairness analysis in requirement assignments to aid negotiation, mediation and decision making. Requirements Engineering, 2009, 14, 231-245.	3.1	83
193	TAIC PART 2007 and Mutation 2007 special issue editorial. Journal of Systems and Software, 2009, 82, 1753-1754.	4.5	0
194	Higher Order Mutation Testing. Information and Software Technology, 2009, 51, 1379-1393.	4.4	216
195	Empirical evaluation of a nesting testability transformation for evolutionary testing. ACM Transactions on Software Engineering and Methodology, 2009, 18, 1-27.	6.0	39
196	Automated Test Data Generation for Coverage: Haven't We Solved This Problem Yet?. , 2009, , .		53
197	Search based data sensitivity analysis applied to requirement engineering. , 2009, , .		33
198	Multi Objective Higher Order Mutation Testing with Genetic Programming. , 2009, , .		30

#	Article	IF	Citations
199	Clustering test cases to achieve effective and scalable prioritisation incorporating expert knowledge. , 2009, , .		133
200	Software project planning for robustness and completion time in the presence of uncertainty using multi objective search based software engineering. , 2009, , .		50
201	Using formal specifications to support testing. ACM Computing Surveys, 2009, 41, 1-76.	23.0	271
202	Improving Web Application Testing using testability measures. , 2009, , .		4
203	Identifying 'Linchpin Vertices' That Cause Large Dependence Clusters. , 2009, , .		13
204	A theoretical and empirical study of EFSM dependence. , 2009, , .		26
205	Measuring and Improving Latency to Avoid Test Suite Wear Out. , 2009, , .		13
206	Control Dependence for Extended Finite State Machines. Lecture Notes in Computer Science, 2009, , 216-230.	1.3	36
207	An empirical study of the relationship between the concepts expressed in source code and dependence. Journal of Systems and Software, 2008, 81, 2287-2298.	4.5	10
208	Special Issue on Searchâ€Based Software Maintenance. Journal of Software: Evolution and Process, 2008, 20, 317-319.	1.1	0
209	Editorial: Testing practice and research. Software Testing Verification and Reliability, 2008, 18, 69-70.	2.0	0
210	Locating dependence structures using search-based slicing. Information and Software Technology, 2008, 50, 1189-1209.	4.4	8
211	Constructing Subtle Faults Using Higher Order Mutation Testing. , 2008, , .		138
212	“Fairness Analysis” in Requirements Assignments. , 2008, , .		36
213	Evaluating Key Statements Analysis. , 2008, , .		3
214	Analysis of Procedure Splitability. , 2008, , .		6
215	Automated Session Data Repair for Web Application Regression Testing. , 2008, , .		27
216	MILU: A Customizable, Runtime-Optimized Higher Order Mutation Testing Tool for the Full C Language. , 2008, , .		108

#	Article	IF	Citations
217	Handling dynamic data structures in search based testing. , 2008, , .		26
218	Dependence Anti Patterns. , 2008, , .		16
219	Search Based Requirements Optimisation: Existing Work and Challenges. , 2008, , 88-94.		75
220	Testability Transformation â€“ Program Transformation to Improve Testability. , 2008, , 320-344.		23
221	Pareto efficient multi-objective test case selection. , 2007, , .		237
222	An empirical study of static program slice size. ACM Transactions on Software Engineering and Methodology, 2007, 16, 8.	6.0	71
223	Empirical study of optimization techniques for massive slicing. ACM Transactions on Programming Languages and Systems, 2007, 30, 3.	2.1	29
224	Search Algorithms for Regression Test Case Prioritization. IEEE Transactions on Software Engineering, 2007, 33, 225-237.	5.6	553
225	Pareto optimal search based refactoring at the design level. , 2007, , .		155
226	A theoretical & empirical analysis of evolutionary testing and hill climbing for structural test data generation. , 2007, , .		66
227	The impact of input domain reduction on search-based test data generation. , 2007, , .		48
228	The Effect of Communication Overhead on Software Maintenance Project Staffing: a Search-Based Approach. , 2007, , .		22
229	Search Based Software Engineering for Program Comprehension. , 2007, , .		28
230	Automated Test Data Generation using Search Based Software Engineering. , 2007, , .		26
231	The Current State and Future of Search Based Software Engineering. , 2007, , .		404
232	A multi-objective approach to search-based test data generation. , 2007, , .		93
233	The multi-objective next release problem. , 2007, , .		159
234	Heuristics for fault diagnosis when testing from finite state machines. Software Testing Verification and Reliability, 2007, 17, 41-57.	2.0	9

#	Article	IF	Citations
235	A non-standard semantics for program slicing and dependence analysis. The Journal of Logic and Algebraic Programming, 2007, 72, 191-206.	1.4	5
236	Equivalence of linear, free, liberal, structured program schemas is decidable in polynomial time. Theoretical Computer Science, 2007, 373, 1-18.	0.9	6
237	Characterising, Explaining, and Exploiting the Approximate Nature of Static Analysis through Animation. , 2006, , .		4
238	Allowing Overlapping Boundaries in Source Code using a Search Based Approach to Concept Binding. , 2006, , .		16
239	Workshop Introduction Astrenet Aspect Analysis. , 2006, , .		0
240	Stop-List Slicing. , 2006, , .		6
241	An Empirical Study of Executable Concept Slice Size. , 2006, , .		6
242	Tool-Supported Refactoring of Existing Object-Oriented Code into Aspects. IEEE Transactions on Software Engineering, 2006, 32, 698-717.	5.6	56
243	Search Based Approaches to Component Selection and Prioritization for the Next Release Problem. Conference on Software Maintenance, Proceedings of the, 2006, , .	0.0	91
244	Search--based approaches to the component selection and prioritization problem. , 2006, , .		13
245	The species per path approach to SearchBased test data generation. , 2006, , .		46
246	Theory and algorithms for slicing unstructured programs. Information and Software Technology, 2006, 48, 549-565.	4.4	17
247	Improving test quality using robust unique input/output circuit sequences (UIOCs). Information and Software Technology, 2006, 48, 696-707.	4.4	9
248	Selected papers from the fourth Source Code Analysis and Manipulation (SCAM 2004) Workshop. Journal of Systems and Software, 2006, 79, 1217-1218.	4.5	1
249	A formalisation of the relationship between forms of program slicing. Science of Computer Programming, 2006, 62, 228-252.	1.9	32
250	A formal relationship between program slicing and partial evaluation. Formal Aspects of Computing, 2006, 18, 103-119.	1.8	7
251	Theoretical foundations of dynamic program slicing. Theoretical Computer Science, 2006, 360, 23-41.	0.9	35
252	Search Based Software Engineering. Lecture Notes in Computer Science, 2006, , 740-747.	1.3	8

#	Article	IF	Citations
253	ConSUS: a light-weight program conditioner. Journal of Systems and Software, 2005, 77, 241-262.	4.5	11
254	Unifying program slicing and concept assignment for higher-level executable source code extraction. Software - Practice and Experience, 2005, 35, 977-1006.	3.6	14
255	Automated Unique Input Output Sequence Generation for Conformance Testing of FSMs. Computer Journal, 2005, 49, 331-344.	2.4	51
256	Generating feasible input sequences for extended finite state machines (EFSMs) using genetic algorithms. , 2005, , .		11
257	An empirical study of the robustness of two module clustering fitness functions. , 2005, , .		62
258	Static Program Slicing Algorithms are Minimal for Free Liberal Program Schemas. Computer Journal, 2005, 48, 737-748.	2.4	14
259	Branch-Coverage Testability Transformation for Unstructured Programs. Computer Journal, 2005, 48, 421-436.	2.4	22
260	Guest Editorial: Special Issue on Software Maintenance and Evolution. IEEE Transactions on Software Engineering, 2005, 31, 801-803.	5.6	1
261	A Survey of Empirical Results on Program Slicing. Advances in Computers, 2004, 62, 105-178.	1.6	100
262	Evolutionary testing in the presence of loop-assigned flags. , 2004, , .		63
263	Syntax-Directed Amorphous Slicing. Automated Software Engineering, 2004, 11, 27-61.	2.9	15
264	CONSIT: a fully automated conditioned program slicer. Software - Practice and Experience, 2004, 34, 15-46.	3.6	23
265	Testability transformation. IEEE Transactions on Software Engineering, 2004, 30, 3-16.	5.6	206
266	Analysis and visualization of predicate dependence on formal parameters and global variables. IEEE Transactions on Software Engineering, 2004, 30, 715-735.	5.6	26
267	How to Overcome the Equivalent Mutant Problem and Achieve Tailored Selective Mutation Using Co-evolution. Lecture Notes in Computer Science, 2004, , 1338-1349.	1.3	81
268	Evolutionary testing in the presence of loop-assigned flags. Software Engineering Notes: an Informal Newsletter of the Special Interest Committee on Software Engineering / ACM, 2004, 29, 108-118.	0.7	17
269	Input Sequence Generation for Testing of Communicating Finite State Machines (CFSMs). Lecture Notes in Computer Science, 2004, , 1429-1430.	1.3	8
270	Amorphous program slicing. Journal of Systems and Software, 2003, 68, 45-64.	4.5	83

#	Article	IF	Citations
271	Equivalence of conservative, free, linear program schemas is decidable. Theoretical Computer Science, 2003, 290, 831-862.	0.9	12
272	Guaranteed inconsistency avoidance during software evolution. Journal of Software: Evolution and Process, 2003, 15, 393-416.	1.1	5
273	An empirical investigation of the influence of a type of side effects on program comprehension. IEEE Transactions on Software Engineering, 2003, 29, 665-670.	5.6	43
274	Search Based Transformations. Lecture Notes in Computer Science, 2003, , 2511-2512.	1.3	12
275	Source code analysis and manipulation. Information and Software Technology, 2002, 44, 717-720.	4.4	0
276	Conditioned slicing supports partition testing. Software Testing Verification and Reliability, 2002, 12, 23-28.	2.0	40
277	Search-based software engineering. Information and Software Technology, 2001, 43, 833-839.	4.4	631
278	Software engineering using metaheuristic innovative algorithms: workshop report. Information and Software Technology, 2001, 43, 905-907.	4.4	15
279	An overview of program slicing. Software Focus, 2001, 2, 85-92.	0.3	119
280	The SEMINAL workshop. Software Engineering Notes: an Informal Newsletter of the Special Interest Committee on Software Engineering / ACM, 2001, 26, 62-66.	0.7	12
281	The Relationship Between Program Dependence and Mutation Analysis. , 2001, , 5-13.		22
282	Espresso. , 2000, , .		9
283	Using program slicing to assist in the detection of equivalent mutants. , 1999, 9, 233-262.		168
284	A new algorithm for slicing unstructured programs. Journal of Software: Evolution and Process, 1998, 10, 415-441.	0.4	28
285	A new algorithm for slicing unstructured programs. , 1998, 10, 415.		3
286	Slicing programs in the presence of errors. Formal Aspects of Computing, 1996, 8, 490-497.	1.8	11
287	A parallel algorithm for static program slicing. Information Processing Letters, 1995, 56, 307-313.	0.6	28
288	Using program slicing to simplify testing. Software Testing Verification and Reliability, 1995, 5, 143-162.	2.0	78

#	Article	IF	Citations
289	Testing of Future Internet Applications Running in the Cloud. Advances in Computer and Electrical Engineering Book Series, 0, , 305-321.	0.3	4