Robert M Hierons

List of Publications by Year
in descending order

188	3,426	279487	205818
papers	citations	23 h-index	48 g-index

200	200	200	1586
all docs	docs citations	times ranked	citing authors

#	Article	IF	Citations
1	An information theoretic notion of software testability. Information and Software Technology, 2022, 143, 106759.	3.0	3
2	Combinatorial testing and modelâ€based testing. Software Testing Verification and Reliability, 2022, 32, .	1.7	1
3	Metamorphic testing and test automation. Software Testing Verification and Reliability, 2022, 32, .	1.7	0
4	Model checking, testing and debugging. Software Testing Verification and Reliability, 2022, 32, .	1.7	0
5	Farewell after an 11â€year journey as joint editorâ€inâ€chief. Software Testing Verification and Reliability, 2022, 32, .	1.7	0
6	$mathcal k$-branching uio sequences for partially specified observable non-deterministic fsms. IEEE Transactions on Software Engineering, 2021, 47, 1029-1040.	4.3	8
7	Using mutual information to test from Finite State Machines: Test suite selection. Information and Software Technology, 2021, 132, 106498.	3.0	6
8	TEA-Cloud: A Formal Framework for Testing Cloud Computing Systems. IEEE Transactions on Reliability, 2021, 70, 261-284.	3.5	16
9	Editorial: Testing, Debugging, and Defect Prediction. Software Testing Verification and Reliability, 2021, 31, e1775.	1.7	0
10	Removing Redundant Refusals: Minimal Complete Test Suites for Failure Trace Semantics. , 2021, , .		0
11	Test case generation for agent-based models: A systematic literature review. Information and Software Technology, 2021, 135, 106567.	3.0	7
12	Minimizing Characterizing sets. Science of Computer Programming, 2021, 208, 102645.	1.5	0
13	Editorial: Verification, reliability and performance. Software Testing Verification and Reliability, 2021, 31, e1790.	1.7	0
14	Adaptive or embedded software testing and mutation testing. Software Testing Verification and Reliability, 2021, 31, .	1.7	1
15	Are 20% of Classes Responsible for 80% of Refactorings?. , 2021, , .		0
16	Mutation Testing for RoboChart. , 2021, , 345-375.		0
17	Property generation/verification and empirical studies. Software Testing Verification and Reliability, 2021, 31, e1800.	1.7	0
18	Efficient state synchronisation in model-based testing through reinforcement learning. , 2021, , .		3

#	Article	IF	Citations
19	Guest Editorial: Special Section on ICTSS. Information and Software Technology, 2020, 118, 106222.	3.0	0
20	Conference Virtualization. Software Testing Verification and Reliability, 2020, 30, e1749.	1.7	0
21	Implementation relations and testing for cyclic systems with refusals and discrete time. Journal of Systems and Software, 2020, 170, 110738.	3.3	6
22	Local Observability and Controllability Analysis and Enforcement in Distributed Testing With Time Constraints. IEEE Access, 2020, 8, 167172-167191.	2.6	4
23	On automation in software engineering. Software Testing Verification and Reliability, 2020, 30, e1753.	1.7	1
24	Working Across Boundaries. Software Testing Verification and Reliability, 2020, 30, e1734.	1.7	0
25	Many-Objective Test Suite Generation for Software Product Lines. ACM Transactions on Software Engineering and Methodology, 2020, 29, 1-46.	4.8	12
26	Inputs and Outputs in CSP. ACM Transactions on Computational Logic, 2020, 21, 1-53.	0.7	6
27	Peer Reviewing in Software Engineering. Software Engineering Notes: an Informal Newsletter of the Special Interest Committee on Software Engineering / ACM, 2020, 45, 18-18.	0.5	2
28	A partial oracle for uniformity statistics. Software Quality Journal, 2019, 27, 1419-1447.	1.4	2
29	Normalised Squeeziness and Failed Error Propagation. Information Processing Letters, 2019, 149, 6-9.	0.4	6
30	Using Squeeziness to test component-based systems defined as Finite State Machines. Information and Software Technology, 2019, 112, 132-147.	3.0	13
31	FSM quasi-equivalence testing via reduction and observing absences. Science of Computer Programming, 2019, 177, 1-18.	1.5	9
32	The world changes. Software Testing Verification and Reliability, 2019, 29, e1723.	1.7	0
33	Testing Robots Using CSP. Lecture Notes in Computer Science, 2019, , 21-38.	1.0	10
34	An Implementation Relation for Cyclic Systems with Refusals and Discrete Time. Lecture Notes in Computer Science, 2019, , 393-409.	1.0	0
35	Local Observability and Controllability Enforcement in Distributed Testing. Communications in Computer and Information Science, 2019, , 327-338.	0.4	0
36	A Mechanised Proof of an Adaptive State Counting Algorithm. Lecture Notes in Computer Science, 2019, , 176-193.	1.0	3

#	Article	IF	Citations
37	A mapping study on testing non-testable systems. Software Quality Journal, 2018, 26, 1373-1413.	1.4	19
38	Bounded Reordering in the Distributed Test Architecture. IEEE Transactions on Reliability, 2018, 67, 522-537.	3.5	30
39	Smart contracts vulnerabilities: a call for blockchain software engineering?. , 2018, , .		127
40	Passive testing with asynchronous communications and timestamps. Distributed Computing, 2018, 31, 327-342.	0.7	23
41	A tool supported methodology to passively test asynchronous systems with multiple users. Information and Software Technology, 2018, 104, 162-178.	3.0	19
42	Introduction to the Software Engineering and Formal Methods 2013 special issue. Software and Systems Modeling, 2017, 16, 5-6.	2.2	0
43	An extended framework for passive asynchronous testing. Journal of Logical and Algebraic Methods in Programming, 2017, 86, 408-424.	0.4	16
44	Testing from Partial Finite State Machines without Harmonised Traces. IEEE Transactions on Software Engineering, 2017, 43, 1033-1043.	4.3	9
45	Implementation relations and probabilistic schedulers in the distributed test architecture. Journal of Systems and Software, 2017, 132, 319-335.	3.3	23
46	Distinguishing Sequences for Distributed Testing: Preset Distinguishing Sequences. Computer Journal, 2017, 60, 110-125.	1.5	2
47	How to manage a conference special issue. Software Testing Verification and Reliability, 2017, 27, e1633.	1.7	0
48	Decidability and complexity for quiescent consistency and its variations. Information and Computation, 2017, 257, 1-21.	0.5	0
49	Parallel Algorithms for Generating Distinguishing Sequences for Observable Non-deterministic FSMs. ACM Transactions on Software Engineering and Methodology, 2017, 26, 1-34.	4.8	11
50	Integrating Graphical and Natural Language Specifications to Support Analysis and Testing. , 2017, , .		1
51	Hardness of Deriving Invertible Sequences from Finite State Machines. Lecture Notes in Computer Science, 2017, , 147-160.	1.0	3
52	Constraint-Based Oracles for Timed Distributed Systems. Lecture Notes in Computer Science, 2017, , 276-292.	1.0	2
53	Decidability and Complexity for Quiescent Consistency. , 2016, , .		4
54	SIP. ACM Transactions on Software Engineering and Methodology, 2016, 25, 1-39.	4.8	81

#	Article	IF	Citations
55	The dreaded desk reject. Software Testing Verification and Reliability, 2016, 26, 3-3.	1.7	2
56	A Suspension-Trace Semantics for CSP. , 2016, , .		6
57	Distinguishing Sequences for Distributed Testing: Adaptive Distinguishing Sequences. Computer Journal, 2016, 59, 1186-1206.	1.5	2
58	Multi-objective optimisation for regression testing. Information Sciences, 2016, 334-335, 1-16.	4.0	33
59	Parallel Algorithms for Generating Harmonised State Identifiers and Characterising Sets. IEEE Transactions on Computers, 2016, 65, 3370-3383.	2.4	14
60	Parallel Algorithms for Testing Finite State Machines:Generating UIO Sequences. IEEE Transactions on Software Engineering, 2016, 42, 1077-1091.	4.3	16
61	Generating Minimum Height ADSs for Partially Specified Finite State Machines. Lecture Notes in Electrical Engineering, 2016, , 171-179.	0.3	0
62	Resolving the Equivalent Mutant Problem in the Presence of Non-determinism and Coincidental Correctness. Lecture Notes in Computer Science, 2016, , 123-138.	1.0	3
63	Controllability Through Nondeterminism in Distributed Testing. Lecture Notes in Computer Science, 2016, , 89-105.	1.0	5
64	Special issue on testing, analysis and debugging of concurrent programs. Software Testing Verification and Reliability, 2015, 25, 165-166.	1.7	0
65	Incomplete Distinguishing Sequences for Finite State Machines. Computer Journal, 2015, 58, 3089-3113.	1.5	14
66	A methodology for validating cloud models using metamorphic testing. Annales Des Telecommunications/Annals of Telecommunications, 2015, 70, 127-135.	1.6	17
67	Generating Complete Controllable Test Suites for Distributed Testing. IEEE Transactions on Software Engineering, 2015, 41, 279-293.	4.3	11
68	A Test Framework for Communications-Critical Large-Scale Systems. IEEE Software, 2015, 32, 86-93.	2.1	0
69	An analysis of the relationship between conditional entropy and failed error propagation in software testing. , 2014, , .		38
70	Automated generation of computationally hard feature models using evolutionary algorithms. Expert Systems With Applications, 2014, 41, 3975-3992.	4.4	15
71	Timed implementation relations for the distributed test architecture. Distributed Computing, 2014, 27, 181-201.	0.7	12
72	Combining Centralised and Distributed Testing. ACM Transactions on Software Engineering and Methodology, 2014, 24, 1-29.	4.8	21

#	Article	IF	Citations
73	Towards estimating computer usersâ€™ mood from interaction behaviour with keyboard and mouse. Frontiers of Computer Science, 2013, 7, 943-954.	1.6	33
74	Semantic mutation testing. Science of Computer Programming, 2013, 78, 345-363.	1.5	18
75	Amorphous Slicing of Extended Finite State Machines. IEEE Transactions on Software Engineering, 2013, 39, 892-909.	4.3	20
76	Verifying and Comparing Finite State Machines for Systems that Have Distributed Interfaces. IEEE Transactions on Computers, 2013, 62, 1673-1683.	2.4	2
77	Using genetic algorithms to generate test sequences for complex timed systems. Soft Computing, 2013, 17, 301-315.	2.1	18
78	Guest Editorial: Special Section from the 11th International Conference on Quality Software (QSIC) Tj ETQq0 0 0 rgBT /Overlock 10 Tf 5	3.0	0
79	Testing Real-Time Embedded Systems using Timed Automata based approaches. Journal of Systems and Software, 2013, 86, 1209-1223.	3.3	23
80	Testing with Inputs and Outputs in CSP. Lecture Notes in Computer Science, 2013, , 359-374.	1.0	12
81	Passive Testing with Asynchronous Communications. Lecture Notes in Computer Science, 2013, , 99-113.	1.0	7
82	An Implementation Relation and Test Framework for Timed Distributed Systems. Lecture Notes in Computer Science, 2013, , 82-97.	1.0	9
83	A Longitudinal Study of Fan-In and Fan-Out Coupling in Open-Source Systems. , 2013, , 199-224.		0
84	Augmenting Sequence Enumeration with String-Rewriting for Requirements Analysis and Behavioral Specification. Lecture Notes in Computer Science, 2013, , 179-193.	1.0	1
85	Complexity of Data Dependence Problems for Program Schemas with Concurrency. ACM Transactions on Computational Logic, 2012, 13, 1-18.	0.7	2
86	Specification Mutation Analysis for Validating Timed Testing Approaches Based on Timed Automata. , 2012, , .		12
87	SMT-C: A Semantic Mutation Testing Tools for C. , 2012, , .		8
88	Oracles for Distributed Testing. IEEE Transactions on Software Engineering, 2012, 38, 629-641.	4.3	36
89	A framework for pathologies of message sequence charts. Information and Software Technology, 2012, 54, 1283-1295.	3.0	5
90	Semantic Mutation Analysis of Floating-Point Comparison. , 2012, , .		6

#	Article	IF	Citations
91	Model Based Test Automation through Asynchronous Channels. , 2012, , .		1
92	The complexity of asynchronous model based testing. Theoretical Computer Science, 2012, 451, 70-82.	0.5	13
93	Using schedulers to test probabilistic distributed systems. Formal Aspects of Computing, 2012, 24, 679-699.	1.4	8
94	Squeeziness: An information theoretic measure for avoiding fault masking. Information Processing Letters, 2012, 112, 335-340.	0.4	27
95	Implementation relations and test generation for systems with distributed interfaces. Distributed Computing, 2012, 25, 35-62.	0.7	32
96	Overcoming controllability problems in distributed testing from an input output transition system. Distributed Computing, 2012, 25, 63-81.	0.7	10
97	Using Time to Add Order to Distributed Testing. Lecture Notes in Computer Science, 2012, , 232-246.	1.0	13
98	Conformance Testing from Message Sequence Charts. , 2011, , .		13
99	Creating adaptive sequences with genetic algorithms to reach a certain state in a non-deterministic FSM. , 2011, , .		1
100	Conformance Relations for Distributed Testing Based on CSP. Lecture Notes in Computer Science, 2011, , 48-63.	1.0	6
101	GeTeX: A Tool for Testing Real-Time Embedded Systems Using CAN Applications. , 2011, , .		5
102	A Longitudinal Study of Fan-In and Fan-Out Coupling in Open-Source Systems. International Journal of Information System Modeling and Design, 2011, 2, 1-26.	0.9	2
103	Decidability of strong equivalence for subschemas of a class of linear, free, near-liberal program schemas. The Journal of Logic and Algebraic Programming, 2011, 80, 92-112.	1.4	2
104	Characterizing minimal semantics-preserving slices of predicate-linear, free, liberal program schemas. The Journal of Logic and Algebraic Programming, 2011, 80, 481-496.	1.4	1
105	An integrated search-based approach for automatic testing from extended finite state machine (EFSM) models. Information and Software Technology, 2011, 53, 1297-1318.	3.0	57
106	Do moods affect programmersâ€™ debug performance?. Cognition, Technology and Work, 2011, 13, 245-258.	1.7	85
107	Testing timed systems modeled by Stream X-machines. Software and Systems Modeling, 2011, 10, 201-217.	2.2	10
108	Scenariosâ€based testing of systems with distributed ports. Software - Practice and Experience, 2011, 41, 999-1026.	2.5	11

#	Article	IF	Citations
109	ICST 2008 Special Issue. Software Testing Verification and Reliability, 2011, 21, 1-1.	1.7	0
110	Automated metamorphic testing on the analyses of feature models. Information and Software Technology, 2011, 53, 245-258.	3.0	62
111	Mutation testing on an object-oriented framework: An experience report. Information and Software Technology, 2011, 53, 1124-1136.	3.0	21
112	On the computational complexity of dynamic slicing problems for program schemas. Mathematical Structures in Computer Science, 2011, 21, 1339-1362.	0.5	2
113	A Case Study on the Use of Genetic Algorithms to Generate Test Cases for Temporal Systems. Lecture Notes in Computer Science, 2011, , 396-403.	1.0	7
114	Experimental Comparison of Different Techniques to Generate Adaptive Sequences. Lecture Notes in Computer Science, 2011, , 404-411.	1.0	1
115	Checking experiments for stream X-machines. Theoretical Computer Science, 2010, 411, 3372-3385.	0.5	6
116	Estimating the feasibility of transition paths inÂextended finite state machines. Automated Software Engineering, 2010, 17, 33-56.	2.2	33
117	Generating a checking sequence with a minimum number of reset transitions. Automated Software Engineering, 2010, 17, 217-250.	2.2	15
118	Canonical finite state machines for distributed systems. Theoretical Computer Science, 2010, 411, 566-580.	0.5	8
119	A trajectory-based strict semantics for program slicing. Theoretical Computer Science, 2010, 411, 1372-1386.	0.5	19
120	Scenarios-Based Testing of Systems with Distributed Ports. , 2010, , .		2
121	Generating Feasible Transition Paths for Testing from an Extended Finite State Machine (EFSM) with the Counter Problem. , 2010, , .		14
122	Semantic Mutation Testing. , 2010, , .		10
123	Reaching and Distinguishing States of Distributed Systems. SIAM Journal on Computing, 2010, 39, 3480-3500.	0.8	19
124	Non-local Choice and Implied Scenarios. , 2010, , .		6
125	Automated Test Data Generation on the Analyses of Feature Models: A Metamorphic Testing Approach. , 2010, , .		28
126	Testing Probabilistic Distributed Systems. Lecture Notes in Computer Science, 2010, , 63-77.	1.0	9

#	Article	IF	Citations
127	Mutation Testing. , 2010, , 594-602.		8
128	Verdict functions in testing with a fault domain or test hypotheses. ACM Transactions on Software Engineering and Methodology, 2009, 18, 1-19.	4.8	19
129	Testing from a stochastic timed system with a fault model. The Journal of Logic and Algebraic Programming, 2009, 78, 98-115.	1.4	32
130	Mutation testing from probabilistic and stochastic finite state machines. Journal of Systems and Software, 2009, 82, 1804-1818.	3.3	47
131	Overcoming controllability problems with fewest channels between testers. Computer Networks, 2009, 53, 680-690.	3.2	7
132	Using status messages in the distributed test architecture. Information and Software Technology, 2009, 51, 1123-1130.	3.0	3
133	Generating Feasible Transition Paths for Testing from an Extended Finite State Machine (EFSM). , 2009, , .		76
134	Simulation Relations for Systems with Distributed Interfaces. , 2009, , .		0
135	Using formal specifications to support testing. ACM Computing Surveys, 2009, 41, 1-76.	16.1	271
136	A Testability Transformation Approach for State-Based Programs. , 2009, , .		4
137	JPEG Steganography: A Performance Evaluation of Quantization Tables. , 2009, , .		20
138	Checking Sequence Construction Using Adaptive and Preset Distinguishing Sequences. , 2009, , .		13
139	Aiding Test Case Generation in Temporally Constrained State Based Systems Using Genetic Algorithms. Lecture Notes in Computer Science, 2009, , 327-334.	1.0	12
140	Using communication coverage criteria and partial model generation to assist software integration testing. Software Quality Journal, 2008, 16, 185-211.	1.4	9
141	Testing a deterministic implementation against a non-controllable non-deterministic stream X-machine. Formal Aspects of Computing, 2008, 20, 597-617.	1.4	4
142	Checking sequences for distributed test architectures. Distributed Computing, 2008, 21, 223-238.	0.7	21
143	Testing in the Distributed Test Architecture: An Extended Abstract. , 2008, , .		2
144	Extending Stream X-Machines to Specify and Test Systems with Timeouts. , 2008, , .		2

#	Article	IF	Citations
145	High Capacity Steganographic Method Based Upon JPEG. , 2008, , .		20
146	Using adaptive distinguishing sequences in checking sequence constructions. , 2008, , .		10
147	Implementation Relations for the Distributed Test Architecture. Lecture Notes in Computer Science, 2008, , 200-215.	1.0	30
148	Testability Transformation â€“ Program Transformation to Improve Testability. , 2008, , 320-344.		23
149	Testing in the Distributed Test Architecture. , 2008, , 157-183.		8
150	Controllable Test Cases for the Distributed Test Architecture. Lecture Notes in Computer Science, 2008, , 201-215.	1.0	17
151	Measuring personality from keyboard and mouse use. , 2008, , .		29
152	A Thread-tag Based Semantics for Sequence Diagrams. , 2007, , .		5
153	The Effect of the Distributed Test Architecture on the Power of Testing. Computer Journal, 2007, 51, 497-510.	1.5	34
154	Mood independent programming. , 2007, , .		12
155	Search Algorithms for Regression Test Case Prioritization. IEEE Transactions on Software Engineering, 2007, 33, 225-237.	4.3	553
156	A Meta-analysis Approach to Refactoring and XP. , 2007, , .		0
157	Mutation Testing from Probabilistic Finite State Machines. , 2007, , .		21
158	Reducing the cost of applying adaptive test cases. Computer Networks, 2007, 51, 224-238.	3.2	3
159	Heuristics for fault diagnosis when testing from finite state machines. Software Testing Verification and Reliability, 2007, 17, 41-57.	1.7	9
160	Mutation Testing from Probabilistic Finite State Machines. , 2007, , .		1
161	Editorial: Addressing industrial challengesâ€"UKTest 2005 and beyond. Software Testing Verification and Reliability, 2006, 16, 131-132.	1.7	0
162	Improving test quality using robust unique input/output circuit sequences (UIOCs). Information and Software Technology, 2006, 48, 696-707.	3.0	9

#	Article	IF	Citations
163	Separating sequence overlap for automated test sequence generation. Automated Software Engineering, 2006, 13, 283-301.	2.2	8
164	Achieving communication coverage in testing. Software Engineering Notes: an Informal Newsletter of the Special Interest Committee on Software Engineering / ACM, 2006, 31, 1-10.	0.5	9
165	ConSUS: a light-weight program conditioner. Journal of Systems and Software, 2005, 77, 241-262.	3.3	11
166	Editorial: Validating our findings. Software Testing Verification and Reliability, 2005, 15, 209-210.	1.7	0
167	Automated Unique Input Output Sequence Generation for Conformance Testing of FSMs. Computer Journal, 2005, 49, 331-344.	1.5	51
168	Generating feasible input sequences for extended finite state machines (EFSMs) using genetic algorithms. , 2005, , .		11
169	Eliminating Redundant Tests in a Checking Sequence. Lecture Notes in Computer Science, 2005, , 146-158.	1.0	21
170	CONSIT: a fully automated conditioned program slicer. Software - Practice and Experience, 2004, 34, 15-46.	2.5	23
171	Editorial: Software testing in the United Kingdom. Software Testing Verification and Reliability, 2004, 14, 165-166.	1.7	0
172	Testing conformance of a deterministic implementation against a non-deterministic stream X-machine. Theoretical Computer Science, 2004, 323, 191-233.	0.5	31
173	How to Overcome the Equivalent Mutant Problem and Achieve Tailored Selective Mutation Using Co-evolution. Lecture Notes in Computer Science, 2004, , 1338-1349.	1.0	81
174	Input Sequence Generation for Testing of Communicating Finite State Machines (CFSMs). Lecture Notes in Computer Science, 2004, , 1429-1430.	1.0	8
175	Conditions for Resolving Observability Problems in Distributed Testing. Lecture Notes in Computer Science, 2004, , 229-242.	1.0	21
176	UIO sequence based checking sequences for distributed test architectures. Information and Software Technology, 2003, 45, 793-803.	3.0	28
177	Editorial: Testing in the large through the small?. Software Testing Verification and Reliability, 2003, 13, 139-140.	1.7	0
178	Concerning the Ordering of Adaptive Test Sequences. Lecture Notes in Computer Science, 2003, , 289-302.	1.0	13
179	Automatically Generating Information from a Z Specification to Support the Classification Tree Method. Lecture Notes in Computer Science, 2003, , 388-407.	1.0	13
180	Editorial: Formal methods and testing. Software Testing Verification and Reliability, 2002, 12, 69-70.	1.7	0

#	Article	IF	Citations
181	Checking states and transitions of a set of communicating finite state machines. Microprocessors and Microsystems, 2001, 24, 443-452.	1.8	10
182	An overview of program slicing. Software Focus, 2001, 2, 85-92.	0.3	119
183	Editorial: special issue on specification-based testing. Software Testing Verification and Reliability, 2000, 10, 201-202.	1.7	14
184	Minimizing the cost of fault location when testing from a finite state machine. Computer Communications, 1999, 22, 120-127.	3.1	3
185	Testing from a Z Specification. Software Testing Verification and Reliability, 1997, 7, 19-33.	1.7	96
186	Estimation of failure rate using random and partition testing. Software Testing Verification and Reliability, 1997, 7, 153-164.	1.7	9
187	A More Precise Implementation Relation for Distributed Testing. Computer Journal, 0, , bxv057.	1.5	1
188	Does an 80:20 rule apply to Java coupling?. , 0, , .		6