

# Papers without the term "property-based testing"

1. DART: directed automated random testing 10.1145/1065010.1065036
2. CUTE: a concolic unit testing engine for C 10.1145/1095430.1081750
3. Feedback-directed Random Test Generation 10.1109/ICSE.2007.37
4. An orchestrated survey of methodologies for automated software test case generation 10.1016/j.jss.2013.02.061
5. JCrasher: an automatic robustness tester for Java 10.1002/spe.602
6. Heuristics for Scalable Dynamic Test Generation 10.1109/ASE.2008.69
7. Grammar-based whitebox fuzzing 10.1145/1375581.1375607
8. Hybrid Concolic Testing 10.1109/ICSE.2007.41
9. A history of Haskell: being lazy with class 10.1145/1238844.1238856
10. Eclat: Automatic generation and classification of test inputs 10.1007/11531142\_22
11. TCS: estimating gene genealogies NÃO TRATA DE PBT
12. Playing by the rules: rewriting as a practical optimisation technique in GHC
13. Automated testing of refactoring engines 10.1145/1287624.1287651
14. Stochastic lambda calculus and monads of probability distributions 10.1145/503272.503288
15. Test Generation Based on Symbolic Specifications 10.1007/978-3-540-31848-4\_1
16. Concolic Testing 10.1145/1321631.1321746
17. The Scalable Commutativity Rule: Designing Scalable Softwarefor Multicore Processors 10.1145/2699681
18. Erlang 10.1145/1810891.1810910
19. Scrap Your Boilerplate With Class: Extensible Generic Functions 10.1145/1090189.1086391
20. Test generation through programming in UDITA 10.1145/1806799.1806835
21. Learn&Fuzz: Machine Learning for Input Fuzzing 10.1109/ASE.2017.8115618
22. Controllable Combinatorial Coverage in Grammar-Based Testing 10.1007/11754008\_2
23. Directed Test Generation using Symbolic Grammars 10.1145/1321631.1321653
24. Scrap more boilerplate: reflection, zips, and generalised casts 10.1145/1016850.1016883
25. Functional Logic Programming 10.1145/1721654.1721675
26. GAST: Generic Automated Software Testing 10.1007/3-540-44854-3\_6
27. Troubleshooting Blackbox SDN Control Software with Minimal Causal Sequences 10.1145/2619239.2626304
28. Debugging Haskell by Observing Intermediate Data Structures
29. Combined Static and Dynamic Automated Test Generation 10.1145/2001420.2001463
30. Taming Compiler Fuzzers 10.1145/2491956.2462173
31. Effective Random Testing of Concurrent Programs 10.1145/1321631.1321679
32. Random testing in Isabelle/HOL
33. Automated Fixing of Programs with Contracts 10.1109/TSE.2014.2312918
34. Decision procedures for algebraic data types with abstractions 10.1145/1707801.1706325
35. Run your research: on the effectiveness of lightweight mechanization 10.1145/2103621.2103691
36. Flask: staged functional programming for sensor networks 10.1145/1411204.1411251
37. Automatic Proof and Disproof in Isabelle/HOL 10.1007/978-3-642-24364-6\_2
38. Intent fuzzer: crafting intents of death 10.1145/2632168.2632169
39. Genetic Algorithms for Randomized Unit Testing 10.1109/TSE.2010.46
40. The power of Pi 10.1145/1411204.1411213
41. Finding errors in .net with feedback-directed random testing 10.1145/1390630.1390643
42. Model-based testing for real 10.1007/s10009-003-0128-3
43. Comparing libraries for generic programming in haskell 10.1145/1543134.1411301
44. Testing monadic code with QuickCheck 10.1145/636517.636527
45. Uniform boilerplate and list processing 10.1145/1291201.1291208
46. E-Assessment as a Service 10.1109/TLT.2010.24
47. EasyCheck — Test Data for Free 10.1007/978-3-540-78969-7\_23
48. Automated Testing and Debugging of SAT and QBF Solvers 10.1007/978-3-642-14186-7\_6
49. Minimization of randomized unit test cases 10.1109/ISSRE.2005.28
50. Constraints as control 10.1145/2103621.2103675
51. An Empirical Study on the Selection of Good Metamorphic Relations 10.1109/COMPSAC.2006.24
52. Fuzzing and delta-debugging SMT solvers 10.1145/1670412.1670413
53. Runtime Verification for Ultra-Critical Systems 10.1007/978-3-642-29860-8\_23
54. An Empirical Comparison of Automated Generation and Classification Techniques for Object-Oriented Unit Testing 10.1109/ASE.2006.13
55. The GUISurfer tool: towards a language independent approach to reverse engineering GUI code 10.1145/1822018.1822045
56. Programming with Arrows 10.1007/11546382\_2
57. Tool support for randomized unit testing 10.1145/1145735.1145741
58. The New Quickcheck for Isabelle 10.1007/978-3-642-35308-6\_10
59. Evolving patches for software repair 10.1145/2001576.2001768
60. LVars: lattice-based data structures for deterministic parallelism 10.1145/2502323.2502326
61. A lightweight approach to datatype-generic rewriting 10.1017/S0956796810000183
62. On theorem prover-based testing 10.1007/s00165-012-0222-y
63. Nighthawk: a two-level genetic-random unit test data generator 10.1145/1321631.1321654
64. Zeno: An Automated Prover for Properties of Recursive Data Structures 10.1007/978-3-642-28756-5\_28
65. Beauty in the beast 10.1145/1291201.1291206
66. On the Predictability of Random Tests for Object-Oriented Software 10.1109/ICST.2008.20
67. Random testing of C calling conventions 10.1145/1085130.1085132
68. Lineage-driven Fault Injection 10.1145/2723372.2723711
69. Haskell program coverage 10.1145/1291201.1291203
70. Haskell program coverage 10.1145/1631687.1596561
71. Combining Testing and Proving in Dependent Type Theory 10.1007/10930755\_12
72. Systematic generation of glass-box test cases for functional logic programs 10.1145/1273920.1273930

73. Increasing Functional Coverage by Inductive Testing: A Case Study 10.1007/978-3-642-16573-3\_10

74. EduComponents: experiences in e-assessment in computer science education 10.1145/1140124.1140150

75. Testing Erlang data types with quviq quickcheck 10.1145/1411273.1411275

76. Hat-Delta---One Right Does Make a Wrong

77. Scaling up automated test generation: Automatically generating maintainable regression unit tests for programs10.1109/ASE.2011.6100059

78. Testing Data Types Implementations from Algebraic Specifications 10.1007/978-3-540-78917-8\_7

79. On the number and nature of faults found by random testing 10.1002/stvr.415

80. Asymptotically almost all  $\lambda$ -terms are strongly normalizing

81. Asymptotic Improvement of Computations over Free Monads 10.1007/978-3-540-70594-9\_20

82. An interactive functional programming tutor 10.1145/2325296.2325356

83. Property Directed Generation of First-Order Test Data

84. Prototyping symbolic execution engines for interpreted languages 10.1145/2541940.2541977

85. Random Testing in PVS

86. Automated Test Generation and Verified Software 10.1007/978-3-540-69149-5\_18

87. Iterative Refinement of Reverse-Engineered Models by Model-Based Testing 10.1007/978-3-642-05089-3\_20

88. Symbolic Test Case Generation for Primitive Recursive Functions 10.1007/978-3-540-31848-4\_2

89. When is a function a fold or an unfold? 10.1016/S1571-0661(04)80906-X

90. Automated specification-based testing of graphical user interfaces

91. Domain-specific languages and code synthesis using Haskell 10.1145/2605205

92. Scrap your nameplate: (functional pearl) 10.1145/1090189.1086389

93. Abstract Diagnosis of Functional Programs 10.1007/3-540-45013-0\_1

94. Contract-driven testing of JavaScript code 10.1007/978-3-642-13953-6\_9

95. Optics in Future Data Center Networks 10.1109/HOTI.2010.10

96. Declarative Debugging with Buddha 10.1007/11546382\_7

97. Chasing Bottoms 10.1007/978-3-540-27764-4\_6

98. DynSem: A DSL for dynamic semantics specification

99. Scheme-based theorem discovery and concept invention 10.1016/j.eswa.2011.06.055

100. Testing and Tracing Lazy Functional Programs Using QuickCheck and Hat 10.1007/978-3-540-44833-4\_3

101. An embedded language framework for hardware compilation

102. Filet-o-Fish: practical and dependable domain-specific languages for OS development 10.1145/1745438.1745446

103. Cause Reduction for Quick Testing 10.1109/ICST.2014.37

104. Integrating Testing and Interactive Theorem Proving 10.4204/EPTCS.70.1

105. Hipster: Integrating Theory Exploration in a Proof Assistant 10.1007/978-3-319-08434-3\_9

106. Minimizing Faulty Executions of Distributed Systems

107. Lazy Assertions 10.1007/978-3-540-27861-0\_1

108. Extended Static Checking by Calculation Using the Pointfree Transform 10.1007/978-3-642-03153-3\_5

109. Rubicon: bounded verification of web applications 10.1145/2393596.2393667

110. Specifying Strategies for Exercises 10.1007/978-3-540-85110-3\_36

111. Relatively complete counterexamples for higher-order programs 10.1145/2813885.2737971

112. Trace-based Reflexive Testing of OO Programs with T2 10.1109/ICST.2008.12

113. A generic library for GUI reasoning and testing 10.1145/1529282.1529307

114. Contracts for Scala 10.1007/978-3-642-16612-9\_5

115. The ACL2 Sedan Theorem Proving System 10.1007/978-3-642-19835-9\_27

116. Boosting the Permissiveness of Dynamic Information-Flow Tracking by Testing 10.1007/978-3-642-33167-1\_4

117. Cause reduction: delta debugging, even without bugs 10.1002/stvr.1574

118. Finding test data with specific properties via metaheuristic search 10.1109/ISSRE.2013.6698888

119. WebSpec: a visual language for specifying interaction and navigation requirements in web applications 10.1007/s00766-011-0124-1

120. An Overview of Edison 10.1016/S1571-0661(05)80546-8

121. Modbat: A Model-Based API Tester for Event-Driven Systems 10.1007/978-3-319-03077-7\_8

122. Verifying Haskell programs by combining testing, model checking and interactive theorem proving 10.1016/j.infsof.2004.07.002

123. Copilot: monitoring embedded systems 10.1007/s11334-013-0223-x

124. MuCheck: an extensible tool for mutation testing of haskell programs 10.1145/2610384.2628052

125. Pragmatic equivalence and safety checking in Cryptol 10.1145/1481848.1481860

126. Establishing flight software reliability: testing, model checking, constraint-solving, monitoring and learning 10.1007/s10472-014-9408-8

127. Towards generic and flexible web services for e-assessment 10.1145/1384271.1384330

128. Towards Systematic Parallel Programming over MapReduce 10.1007/978-3-642-23397-5\_5

129. Compiling Exceptions Correctly 10.1007/978-3-540-27764-4\_12

130. Skeletal program enumeration for rigorous compiler testing 10.1145/3062341.3062379

131. Challenges in Formal Methods for Testing and Verification of Cloud Computing Systems 10.12694/scpe.v16i3.1104

132. Testing and Debugging Techniques for Answer Set Solver Development 10.1017/S1471068410000396

133. Combinatorics of  $\lambda$ -terms: a natural approach 10.1093/logcom/exx018

134. Generating valid grammar-based test inputs by means of genetic programming and annotated grammars 10.1007/s10664-015-9422-4

135. Efficiently Generating Structurally Complex Inputs with Thousands of Objects 10.1007/978-3-540-73589-2\_13

136. State space exploration using feedback constraint generation and Monte-Carlo sampling 10.1145/1287624.1287670

137. Verifying Haskell programs by combining testing and proving 10.1109/QSIC.2003.1319111

138. Automatic Generation of Test Inputs for Mercury 10.1007/978-3-642-00515-2\_6

139. Architecture Aware Parallel Programming in Glasgow Parallel Haskell (GPH) 10.1016/j.procs.2012.04.199

140. Polynomial tuning of multiparametric combinatorial samplers\* 10.1137/1.9781611975062.9

141. Testing and debugging functional reactive programming 10.1145/3110246

142. Field-exhaustive testing 10.1145/2950290.2950336

143. Find a Compiler Bug in 5 Minutes

144. Test-Case Generation for Maude Functional Modules 10.1007/978-3-642-28412-0\_18

145. Calculating with lenses: optimising bidirectional transformations 10.1145/1929501.1929520

146. Automatic Testing of Higher Order Functions 10.1007/11924661\_9

147. QuickFuzz testing for fun and profit 10.1016/j.jss.2017.09.018

148. Palus: a hybrid automated test generation tool for java 10.1145/1985793.1986036
149. The Effect of a Web-based Coding Tool with Automatic Feedback on Students' Performance and Perceptions 10.1145/3159450.3159579
150. Using temporal relations to specify and test an instant messaging server 10.1145/1808266.1808281
151. Combining static analysis and state transition graphs for verification of event-condition-action systems in the RERS 2012 and 2013 challenges 10.1007/s10009-014-0338-x
152. Holistic recommender systems for software engineering 10.1145/2591062.2591081
153. Counting Terms in the Binary Lambda Calculus
154. Engineering software correctness 10.1145/1085114.1085123
155. Random Visual GUI Testing: Proof of Concept
156. On the Applicability of Random Testing for Aspect-Oriented Programs
157. Making Implicit Safety Requirements Explicit 10.1007/978-3-319-10506-2\_6
158. Verifying an Algorithm Computing Discrete Vector Fields for Digital Imaging 10.1007/978-3-642-31374-5\_15
159. Testing type class laws 10.1145/2364506.2364514
160. Application of Domain-aware Binary Fuzzing to Aid Android Virtual Machine Testing 10.1145/2817817.2731198
161. Automated test generation technique for aspectual features in AspectJ 10.1016/j.infsof.2014.05.020
162. On the value of combining feature subset selection with genetic algorithms: faster learning of coverage models 10.1145/1540438.1540456
163. Generics for the working ML'er 10.1145/1292535.1292547
164. An integrated framework for the diagnosis and correction of rule-based programs 10.1016/j.tcs.2010.07.009
165. An Expressive Semantics of Mocking 10.1007/978-3-642-54804-8\_27
166. Using language workbenches and domain-specific languages for safety-critical software development 10.1007/s10270-018-0679-0
167. Adaptive random testing through iterative partitioning revisited 10.1145/1188895.1188903
168. Learning-Based Test Programming for Programmers 10.1007/978-3-642-34026-0\_42
169. Modular specification and verification of a cache-coherent interface 10.1109/FMCAD.2016.7886668
170. Type test scripts for TypeScript testing 10.1145/3133914
171. Boltzmann Samplers for Closed Simply-Typed Lambda Terms 10.1007/978-3-319-51676-9\_8
172. Finding common ground: choose, assert, and assume 10.1145/2338966.2336800
173. A walk through of software testing techniques 10.1109/SYSMART.2016.7894499
174. Meta: Enabling Programming Languages to Learn from the Crowd 10.1145/2984511.2984532
175. Httdp and dmda in the battlefield: a case study in first-year programming instruction 10.1145/1411260.1411262
176. Stream Fusion on Haskell Unicode Strings 10.1007/978-3-642-16478-1\_8
177. A Formal Semantics for the SmartFrog Configuration Language 10.1007/s10922-015-9351-y
178. Multifocal: A Strategic Bidirectional Transformation Language for XML Schemas 10.1007/978-3-642-30476-7\_6
179. Fully Automatic Testing with Functions as Specifications 10.1007/11894100\_2
180. Automated specification analysis using an interactive theorem prover
181. Every Bit Counts: The binary representation of typed data and programs
182. Towards Automatic and Flexible Unit Test Generation for Legacy HPC Code 10.1109/SE-HPCCSE.2016.005
183. Empirical evaluation of the fault detection effectiveness and test effort efficiency of the automated AOP testing approaches 10.1016/j.infsof.2011.05.004
184. Learning refinement types 10.1145/2858949.2784766
185. Programming with Specifications
186. The Effectiveness of T-Way Test Data Generation 10.1007/978-3-540-87698-4\_5
187. An interactive software development workbench based on biomimetic algorithms
188. Evaluating non-adequate test-case reduction 10.1145/2970276.2970361
189. Dynamic witnesses for static type errors (or, ill-typed programs usually go wrong) 10.1145/2951913.2951915
190. Automatically learning shape specifications 10.1145/2908080.2908125
191. Model-Based Test Case Generation for Web Applications 10.1007/978-3-319-09153-2\_19
192. Towards test-driven software defined networking 10.1109/NOMS.2014.6838225
193. BloomUnit: declarative testing for distributed programs 10.1145/2304510.2304512
194. Random Testing: Evaluation of a Law Describing the Number of Faults Found 10.1109/ICST.2012.100
195. Functional pearl: every bit counts 10.1145/1863543.1863548
196. Generating structured test data with specific properties using nested Monte-Carlo search 10.1145/2576768.2598339
197. Model-Based API Testing of Apache ZooKeeper 10.1109/ICST.2017.33
198. A Computational Science Agenda for Programming Language Research 10.1016/j.procs.2014.05.064
199. GUI inspection from source code analysis 10.14279/tuj.eceasst.33.459.447
200. What is the meaning of these constant interruptions? 10.1017/S0956796807006363
201. A generic e-learning multiparadigm programming language system: IDEFIX project 10.1145/611892.612015
202. Find More Bugs with QuickCheck! 10.1109/AST.2016.019
203. A Design for Type-Directed Programming in Java 10.1016/j.entcs.2005.09.014
204. Monadic, Prompt Lazy Assertions in Haskell 10.1007/978-3-540-76637-7\_4
205. Generic generation of elements of types
206. Extending Nunchaku to Dependent Type Theory 10.4204/EPTCS.210.3
207. Balancing weight-balanced trees 10.1017/S0956796811000104
208. Verifying Web Applications: From Business Level Specifications to Automated Model-Based Testing 10.4204/EPTCS.141.2
209. Verification in the Age of Microservices 10.1145/3102980.3102986
210. GUIsurfer: A reverse engineering framework for user interface software
211. Early estimation of defect density using an in-process Haskell metrics model 10.1145/1082983.1083285
212. Model finding for recursive functions in SMT 10.1007/978-3-319-40229-1\_10
213. Flow graphs for testing sequential erlang programs 10.1145/1022471.1022479
214. Systematic synthesis of functions
215. Proof-directed debugging and repair
216. GreenDroid: A tool for analysing power consumption in the android ecosystem 10.1109/Informatics.2015.7377811
217. Test input generation using dynamic programming 10.1145/2393596.2393635
218. Quick specifications for the busy programmer 10.1017/S0956796817000090
219. Connecting a Logical Framework to a First-Order Logic Prover 10.1007/11559306\_17
220. Proving Programs Incorrect Using a Sequent Calculus for Java Dynamic Logic 10.1007/978-3-540-73770-4\_3
221. Type systems for the masses: deriving soundness proofs and efficient checkers 10.1145/2814228.2814239
222. Enumerating Well-Typed Terms Generically 10.1007/978-3-642-11931-6\_5

223. Reasoning with the HERMIT: tool support for equational reasoning on GHC core programs 10.1145/2887747.2804303  
224. Libraries for Generic Programming in Haskell 10.1007/978-3-642-04652-0\_4  
225. The Racket virtual machine and randomized testing 10.1007/s10990-013-9091-1  
226. Randomized Testing in PLT Redex  
227. Finding Inputs that Reach a Target Expression 10.1109/SCAM.2007.30  
228. Generating Random Lambda Calculus Terms  
229. Automated Test Case Generation Based on Coverage Analysis 10.1109/TASE.2009.33  
230. Whispec: white-box testing of libraries using declarative specifications 10.1145/1512762.1512764  
231. Recent Improvements of MagicHaskell 10.1007/978-3-642-11931-6\_9  
232. A Pattern Logic for Prompt Lazy Assertions in Haskell 10.1007/978-3-540-74130-5\_8  
233. Contributing to safety and due diligence in safety-critical interactive systems development by generating and analyzing finite state models 10.1145/1570433.1570474  
234. Software Validation via Model Animation 10.1007/978-3-319-21215-9\_6  
235. Species and functors and types, oh my! 10.1145/1863523.1863542  
236. ACL2 for Freshmen: First Experiences  
237. Automatic Generation of Test Suites from Decision Table - Theory and Implementation 10.1109/ICSEA.2010.78  
238. DoubleCheck your theorems 10.1145/1637837.1637844  
239. Bidirectional Symbolic Analysis for Effective Branch Testing 10.1109/TSE.2015.2490067  
240. A Review of software testing techniques  
241. Mechanized metatheory model-checking 10.1145/1273920.1273931  
242. Data-flow testing of declarative programs 10.1145/1411203.1411233  
243. Automatic testing environment for multi-core embedded software—ATEMES 10.1016/j.jss.2011.08.030  
244. Superposition with Structural Induction 10.1007/978-3-319-66167-4\_10  
245. Testing reactive systems with GAST  
246. An overview of the programatica toolset  
247. T3i: a tool for generating and querying test suites for Java 10.1145/2786805.2803182  
248. Haskell tools from the programatica project 10.1145/871895.871907  
249. A Natural Counting of Lambda Terms 10.1007/978-3-662-49192-8\_15  
250. Random testing for higher-order, stateful programs 10.1145/1932682.1869505  
251. Counting and generating terms in the binary lambda calculus 10.1017/S0956796815000271  
252. Smart Testing of Functional Programs in Isabelle 10.1007/978-3-642-28717-6\_14  
253. Lightweight Automated Testing with Adaptation-Based Programming 10.1109/ISSRE.2012.1  
254. Automated Test Generation and Mutation Testing for Alloy 10.1109/ICST.2017.31  
255. Test Data Generation of Bytecode by CLP Partial Evaluation 10.1007/978-3-642-00515-2\_2  
256. Functional Testing in the Focal Environment 10.1007/978-3-540-79124-9\_7  
257. Fast and sound random generation for automated testing and benchmarking in objective Caml 10.1145/1596627.1596637  
258. Commercial uses: Going functional on exotic trades  
259. TzuYu: Learning stateful tpestates 10.1109/ASE.2013.6693101  
260. True sums of products 10.1145/2633628.2633634  
261. Testing from Algebraic Specifications: Test Data Set Selection by Unfolding Axioms 10.1007/11759744\_14  
262. Axiom-Based Transformations: Optimisation and Testing 10.1016/j.entcs.2009.09.038  
263. Dimensionality reduction and principal surfaces via Kernel Map Manifolds 10.1109/ICCV.2009.5459193  
264. Sneaking around concatMap: efficient combinators for dynamic programming 10.1145/2398856.2364559  
265. Capture and Evolution of Web Requirements Using WebSpec 10.1007/978-3-642-13911-6\_12  
266. Type-safe, self inspecting code 10.1145/1017472.1017485  
267. A lightweight interactive debugger for haskell 10.1145/1291201.1291204  
268. Model-Based Testing for Verification Back-Ends 10.1007/978-3-642-38916-0\_3  
269. Testing .NET Code with YETI 10.1109/ICECCS.2010.58  
270. Refinement reflection: complete verification with SMT 10.1145/3158141  
271. Efficient Exhaustive Generation of Functional Programs Using Monte-Carlo Search with Iterative Deepening 10.1007/978-3-540-89197-0\_21  
272. T3, a Combinator-Based Random Testing Tool for Java: Benchmarking 10.1007/978-3-319-07785-7\_7  
273. Functional reactive programming, refactored 10.1145/3241625.2976010  
274. Uncertainty-Driven Black-Box Test Data Generation 10.1109/ICST.2017.30  
275. Arthur: Rich Post-Facto Debugging for Production Analytics Applications

---

## Research that was not written as an article

276. Towards a practical programming language based on dependent type theory (BOOK)  
277. Concepts, techniques, and models of computer programming (BOOK)  
278. Parallel and concurrent programming in Haskell: Techniques for multicore and multithreaded programming (BOOK)  
279. Exploring Generic Haskell (BOOK)  
280. Practical implementation of a dependently typed functional programming language (THESIS)  
281. Symbolic execution of object oriented programs with axiomatic summaries (PATENT)  
282. A functional specification of effects (THESIS)  
283. Tool support for refactoring Haskell programs (THESIS)  
284. Sat-based finite model generation for higher-order logic (THESIS)  
285. Technological support for distributed agile development (THESIS)  
286. Functional programming and theorem proving for undergraduates: a progress report 10.1145/1411260.1411264 (PROGRESS REPORT)  
287. High assurance programming in Cryptol 10.1145/1558607.1558676 (EXTENDED ABSTRACT)  
288. Model-Based Test Case Generation for Real-Time Systems (THESIS)  
289. Interleaving Strategies 10.1007/978-3-642-22673-1\_14 (TECHNICAL REPORT)  
290. Securing the Automobile: a Comprehensive Approach (TECHNICAL REPORT)  
291. Shrinking and showing functions: (functional pearl) 10.1145/2364506.2364516 (FUNCTIONAL PEARL)

292. (EXPERIENCE REPORT): applying random testing to a base type environment 10.1145/2500365.2500616 (EXPERIENCE REPORT)  
293. Trends in functional programming (BOOK)  
294. Rule-Based Software Verification and Correction (THESIS)  
295. (EXPERIENCE REPORT): a do-it-yourself high-assurance compiler 10.1145/2364527.2364553 (EXPERIENCE REPORT)  
296. Towards Automated Software Testing: Techniques, Classifications and Frameworks (THESIS)  
297. hol-TestGen 1.5. 0 user guide (TECHNICAL REPORT)  
298. Implementing functional logic programs by translation into purely functional programs (DISSERTATION)  
299. Vulnerability modeling and monadic dynamical systems (DISSERTATION)  
300. Implicit self-adjusting computation for purely functional programs (THESIS)  
301. Software Measurement for Functional Programming (THESIS)  
302. (EXPERIENCE REPORT): growing programming languages for beginning students 10.1145/1932681.1863576 (EXPERIENCE REPORT)  
303. An embedded language approach to hardware description and verification (CITATION)  
304. Theories in practice: Easy-to-write specifications that catch bugs (TECHNICAL REPORT)  
305. Stream fusion: practical shortcut fusion for coinductive sequence types (THESIS)  
306. Testing Techniques in Software Engineering (BOOK)  
307. Position paper: the science of deep specification 10.1098/rsta.2016.0331 (POSITION PAPER)  
308. Feedback services for exercise assistants (TECHNICAL REPORT)  
309. Directed random testing (THESIS)  
310. Generic functional programming conceptual design, implementation and applications (BOOK)  
311. Automatic proofs and refutations for higher-order logic (DISSERTATION)  
312. Automatic testing of software with structurally complex inputs (THESIS)

---

**Primary studies that met all criteria**

313. Testing Telecoms Software with Quviq QuickCheck 10.1145/1159789.1159792  
314. Smallcheck and lazy smallcheck: automatic exhaustive testing for small values 10.1145/1411286.1411292  
315. A PropEr integration of types and function specifications with property-based testing 10.1145/2034654.2034663  
316. Finding race conditions in Erlang with QuickCheck and PULSE 10.1145/1631687.1596574  
317. QuickSpec: Guessing Formal Specifications Using Testing 10.1007/978-3-642-13977-2\_3  
318. Automatic testing of TCP/IP implementations using QuickCheck 10.1145/1596600.1596612  
319. Property-Based Testing - The ProTest Project 10.1007/978-3-642-17071-3\_13  
320. Software Testing with QuickCheck 10.1007/978-3-642-17685-2\_6  
321. Ranking programs using black box testing 10.1145/1808266.1808282  
322. Property driven development in Erlang, by example 10.1145/1808266.1808277  
323. Performance Evaluation of A Testing Framework Using QuickCheck and Hadoop 10.2197/ipsjip.20.340  
324. Automatic generation of test models and properties from UML models with OCL constraints 10.1145/2428516.2428525  
325. Turning web services descriptions into quickcheck models for automatic testing 10.1145/2505305.2505306  
326. ArbitCheck: A Highly Automated Property-Based Testing Tool for Java 10.1109/ICSTW.2014.68  
327. Combining Fault-Injection with Property-Based Testing 10.1145/2589650.2559629  
328. Assessing the effects of introducing a new software development process: a methodological description 10.1007/s10009-013-0275-0  
329. PrologCheck – Property-Based Testing in Prolog 10.1007/978-3-319-07151-0\_1  
330. Automating property-based testing of evolving web services 10.1145/2543728.2543741  
331. Improved semantics and implementation through property-based testing with QuickCheck 10.1145/2593501.2593509  
332. SmartCheck: automatic and efficient counterexample reduction and generalization 10.1145/2633357.2633365  
333. Property-based testing of sensor networks 10.1109/SAHCN.2015.7338296  
334. Advanced management of data integrity: property-based testing for business rules 10.1007/s10844-014-0335-2  
335. Foundational Property-Based Testing 10.1007/978-3-319-22102-1\_22  
336. Advances in Lazy SmallCheck 10.1007/978-3-642-41582-1\_4  
337. Automatic Grading of Programming Exercises using Property-Based Testing 10.1145/2899415.2899443  
338. Property-Based Testing with FsCheck by Deriving Properties from Business Rule Models 10.1109/ICSTW.2016.24  
339. FocalTest: A Constraint Programming Approach for Property-Based Testing 10.1007/978-3-642-29578-2\_9  
340. FitSpec: refining property sets for functional testing 10.1145/2976002.2976003  
341. Mysteries of DropBox: Property-Based Testing of a Distributed Synchronization Service 10.1109/ICST.2016.37  
342. Testing Erlang Refactorings with QuickCheck 10.1007/978-3-540-85373-2\_2  
343. Testing AUTOSAR software with QuickCheck 10.1109/ICSTW.2015.7107466  
344. Towards integrating statistical model checking into property-based testing 10.1109/MEMCOD.2016.7797748  
345. Extrapolate: generalizing counterexamples of functional test properties 10.1145/3205368.3205371  
346. Testing properties of weakly consistent programs with Repliss 10.1145/3064889.3064893  
347. Beginner's luck: a language for property-based generators 10.1145/3009837.3009868  
348. A Web Tool for Type Checking and Testing of SPARQL Queries 10.1007/978-3-319-60131-1\_38  
349. Automatic property-based testing and path validation of XQuery programs 10.1002/stvr.1625  
350. Generating good generators for inductive relations 10.1145/3158133  
351. The Cogent Case for Property-Based Testing 10.1145/3144555.3144556  
352. Testing noninterference, quickly 10.1145/2500365.2500574  
353. Failing Faster: Overlapping Patterns for Property-Based Testing 10.1007/978-3-319-51676-9\_7  
354. From Temporal Models to Property-based Testing 10.5220/0006340302410246  
355. Linking unit tests and properties 10.1145/2804295.2804298  
356. Generating constrained random data with uniform distribution 10.1017/S0956796815000143  
357. Effect-driven QuickChecking of compilers 10.1145/3110259  
358. Property-Based Testing with External Test-Case Generators 10.1109/ICSTW.2017.62  
359. Property-based testing for the robot operating system 10.1145/3278186.3278195  
360. Property-based testing for lambda expressions semantics in featherweight Java 10.1145/3264637.3264643

361. AutoBench: comparing the time performance of Haskell programs 10.1145/3242744.3242749  
362. Learning and statistical model checking of system response times 10.1007/s11219-018-9432-8  
363. Generating random well-typed Featherweight Java Programs using quickcheck 10.1016/j.entcs.2019.04.002  
364. ChimpCheck: property-based randomized test generation for interactive apps 10.1145/3133850.3133853  
365. Targeted property-based testing 10.1145/3092703.3092711  
366. Generating Tests by Example 10.1007/978-3-319-73721-8\_19  
367. Property-Based Testing of Browser Rendering Engines with a Consensus Oracle 10.1109/COMPSAC.2018.10270  
368. Property-Based Testing for Spark Streaming 10.1017/S1471068419000012  
369. Show me your properties: the potential of property-based testing in agent-based simulation 10.5555/3374138.3374139  
370. Testing scratch programs automatically 10.1145/3338906.3338910  
371. SolverCheck: Declarative Testing of Constraints 10.1007/978-3-030-30048-7\_33  
372. Property-Based Test Case Generators for Free 10.1007/978-3-030-31157-5  
373. Predicting and Testing Latencies with Deep Learning: An IoT Case Study 10.1007/978-3-030-31157-5\_7  
374. JQF: coverage-guided property-based testing in Java 10.1145/3293882.3339002  
375. Property-based testing of web services by deriving properties from business-rule models 10.1007/s10270-017-0647-0  
376. Coverage guided, property based testing 10.1145/3360607  
377. FlinkCheck: Property-Based Testing for Apache Flink 10.1109/ACCESS.2019.2947361  
378. Genetic algorithms as shrinkers in property-based testing 10.1145/3319619.3322004  
379. Making property-based testing easier to read for humans  
380. Towards the Property-Based Testing of an L4 Microkernel API  
381. Modelling and testing timed data-flow reactive systems in Coq from controlled natural-language requirements  
382. Towards property-based testing of RESTful web services 10.1145/2505305.2505317

---

Obs.: The same research can be excluded due to more than one criterion, but here when excluded by one of them, we do not analyze the others.