

Andrew S Miner

List of Publications by Year in descending order

Source: <https://exaly.com/author-pdf/6098418/publications.pdf>

Version: 2024-02-01

14
papers

243
citations

1307594

7
h-index

1199594

12
g-index

15
all docs

15
docs citations

15
times ranked

111
citing authors

#	ARTICLE	IF	CITATIONS
1	Reachability Set Generation Using Hybrid Relation Compatible Saturation. Lecture Notes in Computer Science, 2020, , 37-51.	1.3	0
2	Improving Saturation Efficiency with Implicit Relations. Lecture Notes in Computer Science, 2019, , 301-320.	1.3	2
3	Computation tree measurement language (CTML). Formal Aspects of Computing, 2018, 30, 443-462.	1.8	4
4	Action and State Based Computation Tree Measurement Language and Algorithms. Lecture Notes in Computer Science, 2018, , 190-206.	1.3	1
5	Automated requirements analysis for a molecular watchdog timer. , 2014, , .		12
6	Explicit State Space and Markov Chain Generation Using Decision Diagrams. Lecture Notes in Computer Science, 2014, , 240-254.	1.3	0
7	Approximate steady-state analysis of large Markov models based on the structure of their decision diagram encoding. Performance Evaluation, 2011, 68, 463-486.	1.2	15
8	A Formal Language toward the Unification of Model Checking and Performance Evaluation. Lecture Notes in Computer Science, 2010, , 130-144.	1.3	3
9	Advanced features in SMART. Performance Evaluation Review, 2009, 36, 58-63.	0.6	19
10	Decision diagrams for the exact solution of Markov models. Proceedings in Applied Mathematics and Mechanics, 2007, 7, 1080701-1080702.	0.2	1
11	Exploiting interleaving semantics in symbolic state-space generation. Formal Methods in System Design, 2007, 31, 63-100.	0.8	25
12	Implicit GSPN reachability set generation using decision diagrams. Performance Evaluation, 2004, 56, 145-165.	1.2	23
13	Using the exact state space of a Markov model to compute approximate stationary measures. Performance Evaluation Review, 2000, 28, 207-216.	0.6	8
14	Efficient Reachability Set Generation and Storage Using Decision Diagrams. Lecture Notes in Computer Science, 1999, , 6-25.	1.3	91