Compiling control

The Journal of Logic Programming
6, 135-162
DOI: 10.1016/0743-1066(89)90033-2

Citation Report

#	Article	IF	Citations
1	Partial evaluation of logic programs. , 1988, , .		0
2	On the transformation of logic programs with instantiation based computation rules. Journal of Symbolic Computation, 1989, 7, 125-154.	0.8	8
3	Compiling bottom-up and mixed derivations into top-down executable logic programs. Journal of Automated Reasoning, 1991, 7, 337-358.	1.4	4
4	Improving the efficiency of constraint logic programming languages by deriving specialized versions. , 1991, , 309-317.		2
5	Efficient implementation of narrowing and rewriting. , 1991, , 344-365.		14
6	On using Eureka properties for transforming generate and test logic programs. , 0, , .		0
7	A general criterion for avoiding infinite unfolding during partial deduction. New Generation Computing, 1992, 11, 47-79.	3.3	58
8	The loop absorption and the generalization strategies for the development of logic programs and partial deduction. The Journal of Logic Programming, 1993, 16, 123-161.	1.7	29
9	Deriving fold/unfold transformations of logic programs using extended OLDT-based abstract interpretation. Journal of Symbolic Computation, 1993, 15, 495-521.	0.8	13
10	Tutorial on specialisation of logic programs. , 1993, , .		113
11	Transformation of logic programs: Foundations and techniques. The Journal of Logic Programming, 1994, 19-20, 261-320.	1.7	143
12	Logic program synthesis. The Journal of Logic Programming, 1994, 19-20, 321-350.	1.7	49
13	Unfolding-definition-folding, in this order, for avoiding unnecessary variables in logic programs. Theoretical Computer Science, 1995, 142, 89-124.	0.9	47
14	Developing correct and efficient logic programs by transformation. Knowledge Engineering Review, 1996, 11, 347-360.	2.6	2
15	Automatic finite unfolding using well-founded measures. The Journal of Logic Programming, 1996, 28, 89-146.	1.7	32
16	Rules and strategies for transforming functional and logic programs. ACM Computing Surveys, 1996, 28, 360-414.	23.0	108
17	Speeding up inferences using relevance reasoning: a formalism and algorithms. Artificial Intelligence, 1997, 97, 83-136.	5.8	26
18	Convergence of program transformers in the metric space of trees. Lecture Notes in Computer Science, 1998, , 315-337.	1.3	8

#	Article	IF	Citations
19	Synthesis and transformation of logic programs using unfold/fold proofs. The Journal of Logic Programming, 1999, 41, 197-230.	1.7	36
20	Convergence of program transformers in the metric space of trees. Science of Computer Programming, 2000, 37, 163-205.	1.9	12
22	Logic program specialisation through partial deduction: Control issues. Theory and Practice of Logic Programming, 2002, 2, 461-515.	1.5	64
23	The List Introduction Strategy for the Derivation of Logic Programs. Formal Aspects of Computing, 2002, 13, 233-251.	1.8	4
24	Schema mediation for large-scale semantic data sharing. VLDB Journal, 2005, 14, 68-83.	4.1	54
25	Inferring non-suspension conditions for logic programs with dynamic scheduling. ACM Transactions on Computational Logic, 2008, 9, 1-43.	0.9	5
26	Efficient generation of test data structures using constraint logic programming and program transformation. Journal of Logic and Computation, 2015, 25, 1263-1283.	0.8	6
27	Abstract conjunctive partial deduction for the analysis and compilation of coroutines. Formal Aspects of Computing, 2017, 29, 125-153.	1.8	2
28	Some thoughts on the role of examples in program transformation and its relevance for explanation-based learning. Lecture Notes in Computer Science, 1989, , 60-77.	1.3	3
29	Implementing finite-domain constraint logic programming on top of a PROLOG-system with delay-mechanism. Lecture Notes in Computer Science, 1990, , 106-117.	1.3	5
30	Improving control of logic programs by using functional logic languages. , 1992, , 1-23.		5
31	Rules and strategies for program transformation. Lecture Notes in Computer Science, 1993, , 263-304.	1.3	7
32	A comparative revisitation of some program transformation techniques. Lecture Notes in Computer Science, 1996, , 355-385.	1.3	15
33	Deriving Transformations of Logic Programs Using Abstract Interpretation. Workshops in Computing, 1993, , 99-117.	0.4	2
34	The Transformational Approach to Program Development. Lecture Notes in Computer Science, 2010, , 112-135.	1.3	3
35	Executing Specifications Using Synthesis and Constraint Solving. Lecture Notes in Computer Science, 2013, , 1-20.	1.3	7
37	The Divide-and-Conquer Subgoal-Ordering Algorithm for Speeding up Logic Inference. Journal of Artificial Intelligence Research, 0, 9, 37-97.	7.0	1
38	Sharing of Computations. DAIMI Report Series, 1993, 22, .	0.1	10

#	Article	IF	Citations
39	Program Derivation = Rules + Strategies. Lecture Notes in Computer Science, 2002, , 273-309.	1.3	7
40	An application of abstract interpretation in source level program transformation. Lecture Notes in Computer Science, 1989, , 35-57.	1.3	1
41	Logic for representing and implementing knowledge about system behaviour. Lecture Notes in Computer Science, 1992, , 42-49.	1.3	1
42	Negation and control in automatically generated logic programs. Lecture Notes in Computer Science, 1992, , 250-264.	1.3	0
43	Synthesis of Narrowing Programs. Workshops in Computing, 1993, , 30-45.	0.4	0
44	Best-first Strategies for Incremental Transformations of Logic Programs. Workshops in Computing, 1993, , 82-98.	0.4	2
45	Using call/exit analysis for logic program transformation. Lecture Notes in Computer Science, 1994, , 36-50.	1.3	0
47	Program Derivation via List Introduction. IFIP Advances in Information and Communication Technology, 1997, , 296-323.	0.7	5
49	Transforming Coroutining Logic Programs into Equivalent CHR Programs. Electronic Proceedings in Theoretical Computer Science, EPTCS, 0, 253, 9-35.	0.8	0
50	Compiling Control as Offline Partial Deduction. Lecture Notes in Computer Science, 2019, , 115-131.	1.3	1
52	From Logic to Functional Logic Programs. Theory and Practice of Logic Programming, 2022, 22, 538-554.	1.5	1
53	Transforming Big-Step toÂSmall-Step Semantics Using Interpreter Specialisation. Lecture Notes in Computer Science, 2023, , 28-38.	1.3	1