Amato, Gianluca; Lipton, James; McGrail, Robert
On the algebraic structure of declarative programming languages. (English) Zbl 1187.68121

Summary: We develop an algebraic framework, Logic Programming Doctrines, for the syntax, proof theory, operational semantics and model theory of Horn Clause logic programming based on indexed premonoidal categories. Our aim is to provide a uniform framework for logic programming and its extensions capable of incorporating constraints, abstract data types, features imported from other programming language paradigms and a mathematical description of the state space in a declarative manner. We define a new way to embed information about data into logic programming derivations by building a sketch-like description of data structures directly into an indexed category of proofs. We give an algebraic axiomatization of bottom-up semantics in this general setting, describing categorical models as fixed points of a continuous operator.

MSC:
68N17 Logic programming

Keywords:
categorical logic; indexed categories; logic programming; abstract data types; constraint logic programming

Software:
Hiord; HiLog; Haskell; CCSL

Full Text: DOI

References:
[2] Amato, G.; Lipton, J., Indexed categories and bottom-up semantics of logic programs, (), 438-454 · Zbl 1275.68042
[5] Bainbridge, E.S.; Scedrov, A.; Freyd, P.J.; Scott, P.J., Functorial polymorphism, (), 315-327, (chapter 14) · Zbl 0717.18005
[13] Cheney, J.; Urban, C., @$\$alpha$@#$-prolog: A logic programming language with names, binding and @$\$alpha$@#$-equivalence, (), 269-283 · Zbl 1104.68370
[16] Corradini, A.; Asperti, A., A categorical model for logic programs: indexed monoidal categories, (), 110-137
[18] Cortesi, A.; Filé, G.; Winsborough, W.W., Prop revisited: proposition al formula as abstract domain for groundness analysis, (), 322-327
[19] Cousot, P.; Cousot, R., Abstract interpretation: A unified lattice model for static analysis of programs by construction or approximation of fixpoints, (), 238-252
[20] Cousot, P.; Cousot, R., Systematic design of program analysis frameworks, (), 269-282 · Zbl 0788.68004
[23] Finkelstein, S.E.; Freyd, P.J.; Lipton, J., Logic programming in tan categories, (), 249-263 · Zbl 1044.68538
[25] Freyd, P.J.; Scedrov, A., Categories, allegories, (1990), North Holland · Zbl 0698.18007
[31] Gorlenko, J.A.; Meseguer, J., Eqlog: equality, types, and generic modules for logic programming, (), 295-363
[32] Hagino, T., A typed lambda calculus with categorical type constructors, (), 140-157
[34] C. Hermida, Fibrations, logical predicates and related topics, Ph.D. Thesis, University of Edinburgh, 1993
[35] Hodas, J.S.; Miller, D., Logic programming in a fragment of intuitionistic linear logic, Information and computation, 110, 2, Zbl 0807.68016
[38] Jaffar, J.; Stuckey, P.J., Semantics of infinite tree logic programming, Theoretical computer science, 46, 2-3, 141-158, (1986) · Zbl 0621.68054
[40] Jeavons, J.S., An alternative linear semantics for allowed logic programs, Annals of pure and applied logic, 84, 1, 3-16, (1997) · Zbl 0876.68023
[41] Kinoshita, Y.; Power, J., A fibrational semantics for logic programs, (), 177-192
[42] Kock, A.; Reyes, G.E., Doctrines in categorical logic, (), 283-313
[44] Lambek, J., From lambda calculus to Cartesian closed categories, (), 376-402
[45] Lambek, J.; Scott, P.J., ()
[47] Lawvere, F.W., Equality in hyperdoctrines and comprehension schema as an adjoint functor, (), 1-14 · Zbl 0234.18002
[50] Leifer, J.J.; Milner, R., Deriving bisimulation congruences for reactive systems, (), 243-258 · Zbl 0999.68141
[51] Lipton, J.; McGrail, R.W., Encapsulating data in logic programming via categorical constraints, (), 391-410 · Zbl 0947.03045
[53] Mac Lane, S., Categories for the working Mathematician, Graduate texts in mathematics, vol. 5, (1971), Springer · Zbl 0945.68020
[58] Makkai, M.; Reyes, G.E., ()
[67] Panangaden, P.; Saraswat, V.A.; Scott, P.J.; Seely, R.A.G., A hyperdoctrinal view of concurrent constraint programming, (), 457-476
[68] Peyton Jones, S.L., ()
[72] Meta-programming in logic programming, () · Zbl 0900.68137
[75] Saraswat, V.A., The category of constraint systems is Cartesian-closed, (), 341-345
[78] Wagner, E.G., Categories, data types and imperative languages, in: Pitt et al. [70], pp. 143-162

This reference list is based on information provided by the publisher or from digital mathematics libraries. Its items are heuristically matched to zbMATH identifiers and may contain data conversion errors. It attempts to reflect the references listed in the original paper as accurately as possible without claiming the completeness or perfect precision of the matching.