

# Bibliography

- [1] B. A and D. M. Berry. Operational and Denotational Semantics of Prolog. *Journal of Logic Programming*, 4:309–330, 1987.
- [2] S. Abiteboul and S. Grumbach. A Rule Based Language with Functions and Sets. Technical Report Technical Report, Institut National de Recherche en Informatique et en Automatique, Le Chesnay, 1990.
- [3] G. Abowd, R. Allan, and D. Garlan. Formalizing style to understand descriptions of software architecture. *ACM Transactions on Software Engineering and Methodology*, 4(4):319–364,, October 1995.
- [4] S. Abramsky. Computational interpretations of linear logic. *Theoretical Computer Science*, 1992.
- [5] S. Abramsky and C. Hankin. *Abstract Interpretation of Declarative Languages*. Ellis Horwood Ltd, 1987.
- [6] S. Abramsky and C. Hankin. An Introduction to Abstract Interpretation. In S. Abramsky and C. Hankin, editors, *Abstract Interpretation of Declarative Languages*, pages 9–31. Ellis Horwood Ltd, 1987.
- [7] H. Ait-Kaci and R. Nasr. A logic programming language with built-in inheritance. *Journal of Logic Programming*, 3:185–215, 1986.
- [8] P. Alexander, P. Baraona, , and J. Penix. Using declarative specifications and case-based planning for system synthesis. *Concurrent Engineering: Research and Applications*, 2(4):291–302, 1994.
- [9] R. Allan and D. Garlan. Formalizing architectural connection. In *Proceedings of the 16th International Conference on Software Engineering*, pages 71–80, 1994.
- [10] M. Alpuente and M. Falaschi. Narrowing as an Incremental Constraint Satisfaction Algorithm. In J. Maluszyński and M. Wirsing, editors, *Proc. of PLILP'91*, volume 528 of *Lecture Notes in Computer Science*, pages 111–122. Springer-Verlag, 1991.
- [11] M. Alpuente, M. Falaschi, M. Gabbrielli, and G. Levi. The semantics of equational logic programming as an instance of CLP. In K. R. Apt, J. W. de Bakker, and J. J. M. M. Rutten, editors, *Logic Programming Languages: Constraints, Functions and Objects*, pages 49–81. The MIT Press, 1993.
- [12] G. Amato and G. Levi. Properties of the lattice of observables in logic programming. In M. Falaschi and M. Navarro, editors, *Proceedings of the APPIA-GULP-PRODE'97 Joint Conference on Declarative Programming*, 1997.
- [13] K. R. Apt. Introduction to Logic Programming. In J. van Leeuwen, editor, *Handbook of Theoretical Computer Science*, volume B: Formal Models and Semantics, pages 495–574. Elsevier and The MIT Press, 1990.
- [14] K. R. Apt, M. Gabbrielli, and D. Pedreschi. A closer look to declarative interpretations. *Journal of Logic Programming*, 28(2):147–180, 1996.
- [15] K. R. Apt and E. Marchiori. Reasoning about Prolog programs: from modes through types to assertions. *Formal Aspects of Computing*, 6(6A):743–765, 1994.
- [16] K. R. Apt and D. Pedreschi. Reasoning about termination of pure PROLOG programs. *Information and Computation*, 106(1):109–157, 1993.

- [17] K. R. Apt and M. H. van Emden. Contributions to the theory of logic programming. *Journal of the ACM*, 29(3):841–862, 1982.
- [18] A. Asperti and S. Martini. Projections instead of variables, a category theoretic interpretation of logic programs. In *Proc. 6<sup>th</sup> ICLP*, pages 337–352. MIT Press, 1989.
- [19] Jean-Pierre Bantre and Daniel Le Mtayer. Programming by Multiset Transformation. *Communications of the ACM*, 36(1):98–111, January 1993.
- [20] Jean-Pierre Bantre and Daniel Le Mtayer. Gamma and the chemical reaction model: ten years after. In *Coordination programming: mechanisms, models and semantics*, pages 3–41. World Scientific Publishing, IC Press, 1996.
- [21] P. Baraona, J. Penix, and P. Alexander. Vspec: A declarative requirements specification language for vhdl. In *Current Issues in Electronic Modeling, Special Issue on Specification Languages and HDLs*, volume 3, pages 51–75, 1995.
- [22] R. Barbuti, M. Codish, R. Giacobazzi, and G. Levi. Modelling Prolog Control. In *Proc. Nineteenth Annual ACM Symp. on Principles of Programming Languages*, pages 95–104. ACM Press, 1992.
- [23] R. Barbuti, R. Giacobazzi, and G. Levi. A Declarative Abstract Semantics for Logic Programs. In A. Bertoni, C. Böhm, and P. Miglioli, editors, *Proc. of the Third Italian Conference on Theoretical Computer Science*, pages 84–96. World Scientific, 1989.
- [24] R. Barbuti, R. Giacobazzi, and G. Levi. A general framework for semantics based, bottom-up abstract interpretation of logic programs. *ACM Transactions on Programming Languages and Systems*, 15(1):133–181, 1993.
- [25] Michael Barr. Models of horn theories. In *Proceedings of Categories in Computer Science and Logic*, 1987.
- [26] D. Batory and S. O’Malley. The design and implementation of hierarchical software systems with reusable components. *ACM Transactions on Software Engineering and Methodology*, 1:355–398, 1992.
- [27] M. Bellia, E. Dameri, P. Degano, G. Levi, and M. Martelli. A formal model for lazy implementations of a PROLOG compatible functional language. In J. A. Campbell, editor, *Issues in PROLOG Implementations*, pages 309–326. Ellis-Horwood, 1984.
- [28] M. Bellia and G. Levi. The relation between logic and functional languages. *Journal of Logic Programming*, 3:217–236, 1986.
- [29] G. Berry and G. Boudol. The chemical abstract machine. In *Proc. Seventeenth Annual ACM Symp. on Principles of Programming Languages*, pages 81–94, 1990.
- [30] G. Berry and G. Boudol. The chemical abstract machine. In *Proceedings of the 17th Annual Symposium of Programming Languages*, pages 81–94, 1990.
- [31] G. Berry and G. Boudol. The chemical abstract machine. *Theoretical Computer Science*, 96:217–248, 1992.
- [32] E. Bertino, M. Martelli, and D. Montesi. CLP(AD) as a Deductive Database Language with Updates. In E. Lamma and P. Mello, editors, *Extensions of Logic Programming. Proc. Third International Workshop on Extensions of Logic Programming, ELP’92*, volume 660 of *Lecture Notes in Artificial Intelligence*, pages 80–99. Springer-Verlag, 1992.
- [33] E. Best, F.S. de Boer, and C. Palamidessi. Partial Order and SOS Semantics for Linear Constraint Programs. In D. Garlan and D. Le Métayer, editors, *Proc. of Coordination 97*, volume 1282 of *Lecture Notes in Computer Science*, pages 256–273. Springer-Verlag, 1997.
- [34] Eike Best, Frank S. de Boer, and Catuscia Palamidessi. Partial order and sos semantics for linear constraint programs. In D. Garlan and D. Le Métayer, editors, *Proc. of Coordination 97*, volume 1282 of *Lecture Notes in Computer Science*, pages 256–273. Springer-Verlag, 1997.

- [35] Eike Best and Catuscia Palamidessi. Linear constraint systems as high-level nets. In U. Montanari and V. Sassone, editors, *Proc. of CONCUR 96*, volume 1119 of *Lecture Notes in Computer Science*, pages 498–513. Springer-Verlag, 1996.
- [36] R. S. Bird and O. De Moor. List partitions. *Formal Aspects of Computing*, 5(1):61–78, 1993.
- [37] R. S. Bird and O. De Moor. *Algebra of Programming*, volume 100 of *International Series in Computer Science*. Prentice Hall, 1996.
- [38] Richard F. Blute. Hyperdoctrines for linear logic. Manuscript.
- [39] E. Börger. A logical operational semantics of full prolog. In E. Börger, H. Kleine, H. Büning, and M. Richter, editors, *CSL 89. 3rd workshop on Computer Science Logic*, volume 440 of *Lecture Notes in Computer Science*, pages 36–64. Springer-Verlag, 1990.
- [40] E. Börger and E. Riccobene. A formal specification of prolog. In M. Droste and Y. Gurevich, editors, *Semantics of Programming Languages and Model Theory*. 1992.
- [41] A. Bossi, M. Gabbrielli, G. Levi, and M. Martelli. The s-semantics approach: Theory and applications. *Journal of Logic Programming*, 19–20:149–197, 1994.
- [42] A. Bossi, M. Gabrielli, G. Levi, and M. C. Meo. A compositional semantics for logic programs. *Theoretical Computer Science*, 122(1-2):3–47, 1994.
- [43] Gerard Boudol. Asynchrony and the  $\pi$ -calculus. Technical Report 1702, INRIA, Sophia-Antipolis, 1992.
- [44] D. Boulanger, M. Bruynooghe, and M. Denecker. Abstracting s-semantics: A model-theoretic approach. In M. Hermenegildo and J. Penjam, editors, *Programming Language Implementation and Logic Programming, Proc. PLILP'94*, Lecture Notes in Computer Science. Springer-Verlag, 1994.
- [45] Pascal Brisset. Implementation d'un langage de programmation logique d'ordre superieur avec MALI. Technical Report 1119, INRIA-Rennes, November 1989.
- [46] M. Bruynooghe. A Practical Framework for the Abstract Interpretation of Logic Programs. *Journal of Logic Programming*, 10(2):91–124, 1991. Revised version of K.U.L. technical report CW-62, 1987.
- [47] F. Bry. Query Evaluation in Recursive Databases: Bottom-up and Top-down Reconciled. In W. Kim, J. M. Nicolas, and S. Nishio, editors, *Proc. first International Conference on Deductive and Object-Oriented Databases*, pages 25–44. Elsevier Science Publishers B. V. (North-Holland), 1990.
- [48] Michele Bugliesi, Giorgio Delzanno, Luigi Liquori, and Maurizio Martelli. A linear logic calculus of objects. In M. Maher, editor, *Proceedings of the Joint International Conference and Symposium on Logic Programming*. MIT Press, September 1996.
- [49] R.M. Burstall and D.E. Rydeheard. *Computational Category Theory*. Prentice-Hall, 1988.
- [50] Iliano Cervesato and Frank Pfenning. A linear logic framework. In *Proceedings, Eleventh Annual IEEE Symposium on Logic in Computer Science*, pages 264–275, New Brunswick, New Jersey, July 1996. IEEE Computer Society Press.
- [51] D. Chan. Constructive Negation Based on the Completed Database. In R. A. Kowalski and K. A. Bowen, editors, *Proc. Fifth Int'l Conf. on Logic Programming*, pages 111–125. The MIT Press, 1988.
- [52] Emily Chapman. Compilation of Logic Programs to a Relational Machine. Honor's Thesis, Wesleyan University, 1997.
- [53] Emily Chapman and James Lipton. Some notes on logic programming with a relational machine. In *Using Relational Methods in Computer Science*. Institute for Software Technology, Universität Bundeswehr, München, 1998.
- [54] Jawahar Chirimar. *Proof Theoretic Approach to Specification Languages*. PhD thesis, University of Pennsylvania, 1995.
- [55] Jawahar Chirimar. *Proof Theoretic Approach to Specification Languages*. PhD thesis, University of Pennsylvania, February 1995.

- [56] K. L. Clark. Negation as Failure. In H. Gallaire and J. Minker, editors, *Logic and Data Bases*, pages 293–322. Plenum Press, New York, 1978.
- [57] K. L. Clark. Predicate logic as a computational formalism. Res. Report DOC 79/59, Imperial College, Dept. of Computing, London, 1979.
- [58] C. Codognet, P. Codognet, and M. Corsini. Abstract Interpretation for Concurrent Logic Languages. In S. K. Debray and M. Hermenegildo, editors, *Proc. North American Conf. on Logic Programming'90*, pages 215–232. The MIT Press, 1990.
- [59] Jacques Cohen. Constraint logic programming. *Communications of the ACM*, 33(7), July 1990.
- [60] M. Comini, G. Levi, M. C. Meo, and G. Vitiello. Proving properties of logic programs by abstract diagnosis. In M. Dams, editor, *Analysis and Verification of Multiple-Agent Languages, 5th LOMAPS Workshop*, number 1192 in Lecture Notes in Computer Science, pages 22–50. Springer-Verlag, 1996.
- [61] M. Comini and M. C. Meo. Compositionality properties of SLD-derivations. *Theoretical Computer Science*, 1997. To appear.
- [62] A. Corradini and A. Asperti. A categorical model for logic programs: Indexed monoidal categories. In *Proceedings REX Workshop '92*. Springer Lecture Notes in Computer Science, 1992.
- [63] A. Corradini and U. Montanari. An algebraic semantics of logic programs as structured transition systems. In *Proceedings of the North American Conference on Logic Programming (NACLP '90)*. MIT Press, 1990.
- [64] Andrea Corradini and Ugo Montanari. An algebraic semantics for structured transition systems and its application to logic programs. *Theoretical Computer Science*, 103:51–106, 1992.
- [65] P. Cousot and R. Cousot. Abstract Interpretation and Applications to Logic Programs. *Journal of Logic Programming*, 13(2–3):103–179, 1992.
- [66] Roy Crole. *Categories for Types*. Cambridge University Press, 1993.
- [67] F. S. de Boer, M. Gabbrielli, E. Marchiori, and C. Palamidessi. Proving concurrent constraint programs correct. In *Proceeding of the twentieth ACM Symposium on Principles of Programming Languages*, pages 98–108. ACM, 1994.
- [68] F.S. de Boer and C. Palamidessi. Embedding as a Tool for Language Comparison. *Information and Computation*, 108(1):128–157, 1994.
- [69] F.S. de Boer, A. Di Pierro, and C. Palamidessi. Infinite computations in nondeterministic constraint programming. *Theoretical Computer Science*, 151(1):37–78, 1995.
- [70] F.S. de Boer, A. Di Pierro, and C. Palamidessi. An algebraic perspective of constraint logic programming. *Journal of Logic and Computation*, 7(1):1–38, 1997.
- [71] O. De Moor. Categories, relations and dynamic programming. *Mathematical Structures in Computing Science*, 4:33–69, 1994.
- [72] O. De Moor. A generic program for sequential decision processes. In M. Hermenegildo and D. S. Swierstra, editors, *Programming Languages: Implementations, Logics, and Programs*, volume 982 of *Lecture Notes in Computer Science*. Springer-Verlag, 1995. Invited Paper.
- [73] Oege de Moor. *Categories, Relations, and Dynamic Programming*. PhD thesis, St. John's College, Oxford, 1991.
- [74] S. K. Debray and P. Mishra. Denotational and Operational Semantics for Prolog. *Journal of Logic Programming*, 5:61–91, 1988.
- [75] Giorgio Delzanno and Maurizio Martelli. Objects in forum. In *Proceedings of the International Logic Programming Symposium*, 1995.
- [76] F. Denis and J. P. Delahaye. Unfolding, Procedural and Fixpoint Semantics of Logic Programs. In C. Choffrut and M. Jantzen, editors, *STACS 91*, volume 480 of *Lecture Notes in Computer Science*, pages 511–522. Springer-Verlag, 1991.

- [77] R. Diaconescu. *Category Semantics for Equational and Constraint Logic Programming*. PhD thesis, Oxford University, 1994.
- [78] R. D. Diaconescu and J. Goguen. An introduction to category-based equational logic. In *Proceedings of TAPSOFT '95*, 1995.
- [79] Scott Dietzen and Frank Pfenning. Higher-order and modal logic as a framework for explanation-based generalization. In Alberto Maria Segre, editor, *Sixth International Workshop on Machine Learning*, pages 447–449, San Mateo, California, June 1989. Morgan Kaufmann Publishers. Expanded version available as Technical Report CMU-CS-89-160, Carnegie Mellon University, Pittsburgh.
- [80] Scott Dietzen and Frank Pfenning. A declarative alternative to assert in logic programming. In *Proceedings of the 1991 International Logic Programming Symposium*, pages 372–386. MIT Press, 1991.
- [81] D. Dougherty and P. Johann. An improved general  $E$ -unification method. In M. Stickel, editor, *Proceedings 10th International Conference on Automated Deduction, Kaiserslautern (Germany)*, volume 449 of *Lecture Notes in Computer Science*, pages 261–275. Springer-Verlag, July 1990.
- [82] D. J. Dougherty. Higher-order unification via combinators. *Theoretical Computer Science*, 114(2):273–298, June 1993.
- [83] D. J. Dougherty and P. Johann. An improved general  $E$ -unification method. *J. Symbolic Computation*, 14(4):303–320, October 1992.
- [84] Dan Dougherty and Pierre Lescanne. Standardization and strong normalization in calculi of explicit substitutions. 1997.
- [85] Daniel Dougherty. Adding algebraic rewriting to the untyped lambda calculus (extended abstract). In Ron Book, editor, *Proceedings of the Fourth International Conference on Rewriting Techniques and Applications (Como, Italy)*, volume 488 of *Lecture Notes in Computer Science*, pages 37–48, Berlin, April 1991. Springer Verlag.
- [86] Daniel J. Dougherty. Adding algebraic rewriting to the untyped lambda calculus. *Information and Computation*, 101(2):251–267, December 1992.
- [87] Daniel J. Dougherty. Some lambda calculi with categorical sums and products. In *Proc. Rewriting Techniques and Applications*, pages 137–151, 1993.
- [88] Daniel J. Dougherty and Patricia Johann. A combinatory logic approach to higher-order  $E$ -unification. *Theoretical Computer Science*, 139:207–242, 1995.
- [89] Daniel J. Dougherty and Ramesh Subrahmanyam. Equality between functionals in the presence of co-products. In *10th Symposium on Logic in Computer Science*, pages 282–291, 1995.
- [90] Conal Elliott and Frank Pfenning. eLP, a Common Lisp Implementation of  $\lambda$ Prolog. Implemented as part of the CMU ERGO project, May 1989.
- [91] M. Falaschi, M. Gabbrielli, K. Marriott, and C. Palamidessi. Confluence in concurrent constraint programming. *Theoretical Computer Science*, 183(2):281–315, 1997.
- [92] M. Falaschi, G. Levi, M. Martelli, and C. Palamidessi. Declarative Modeling of the Operational Behavior of Logic Languages. *Theoretical Computer Science*, 69(3):289–318, 1989.
- [93] M. Falaschi, G. Levi, M. Martelli, and C. Palamidessi. A Model-Theoretic Reconstruction of the Operational Semantics of Logic Programs. *Information and Computation*, 102(1):86–113, 1993.
- [94] Moreno Falaschi, Maurizio Gabbrielli, Kim Marriott, and Catuscia Palamidessi. Compositional analysis for concurrent constraint programming. In M. Vardi, editor, *Eighth Annual Symposium on Logic in Computer Science*, pages 210–221. IEEE, June 1993.
- [95] Moreno Falaschi, Maurizio Gabbrielli, Kim Marriott, and Catuscia Palamidessi. Constraint Logic Programming with Dynamic Scheduling: A Semantics Based on Closure Operators. *Information and Computation*, 137(1):41–67, 1997.

- [96] Amy Felty. *Specifying and Implementing Theorem Provers in a Higher-Order Logic Programming Language*. PhD thesis, University of Pennsylvania, August 1989.
- [97] Amy Felty. Defining object-level parsers in  $\lambda$ prolog. In *Proceedings of the Workshop on the  $\lambda$ Prolog Programming Language*, 1992. Department of Computer and Information Science, University of Pennsylvania, Technical Report MS-CIS-92-86.
- [98] Amy Felty. A logic programming approach to implementing higher-order term rewriting. In Lars-Henrik Eriksson, Lars Hallnäs, and Peter Schroeder-Heister, editors, *Proceedings of the January 1991 Workshop on Extensions to Logic Programming*, pages 135–161. Springer-Verlag Lecture Notes in Artificial Intelligence, 1992.
- [99] Amy Felty and Dale Miller. Specifying theorem provers in a higher-order logic programming language. In *Ninth International Conference on Automated Deduction*, pages 61–80, Argonne, IL, May 1988. Springer-Verlag.
- [100] Stacy Finkelstein, Peter Freyd, and James Lipton. A new framework for declarative programming. *Theoretical Computer Science*.
- [101] Stacy E. Finkelstein. *Tau Categories and Logic Programming*. PhD thesis, University of Pennsylvania, 1994.
- [102] Stacy E. Finkelstein, Peter Freyd, and James Lipton. Logic programming in tau categories. In *Computer Science Logic '94, LNCS 933*. Springer, 1995.
- [103] M. Fitting. A Kripke-Kleene semantics for logic programs. *Journal of Logic Programming*, 2:295–312, 1985.
- [104] Melvin Fitting. A deterministic prolog fixpoint semantics. *Journal of Logic Programming*, 2(2):111–118, 1985.
- [105] R. W. Floyd. Assigning Meanings to Programs. In J. T. Schwartz, editor, *Symp. in Applied Mathematics of the AMS*, pages 19–32, 1967.
- [106] Cédric Fournet and George Gonthier. The reflexive cham and the join-calculus. In *Int'l Conference on Principles of Programming Languages*, pages 372–385. ACM, 1996.
- [107] P. Freyd, P. Mulry, G. Rosolini, and D. Scott. Extensional pers. *Theoretical Computer Science*, 1993.
- [108] Peter Freyd. The logic of cartesian categories. To appear.
- [109] Peter Freyd. On canonizing category theory or on functorializing model theory. A Pamphlet, March 1974.
- [110] Peter Freyd. Remarks on algebraically compact categories. In *Proceedings of the LMS Symposium, Durham, 1991*, volume 177. London Mathematical Society Lecture Note Series, Cambridge University Press, 1992.
- [111] Peter Freyd and Andre Scedrov. Some semantic aspects of polymorphic lambda calculus. *TCS*, 1989. An earlier version appeared in Proc. 2nd IEEE Symposium Logic in Computer Science, Ithaca, N.Y., 1987.
- [112] Peter Freyd and Andre Scedrov. *Categories, Allegories*. North-Holland, 1990.
- [113] M. Gabbrielli, G. M. Dore, and G. Levi. Observable semantics for constraint logic programs. *J. Logic Computation*, 5(2):133–171, 1995.
- [114] J. H. Gallier and S. Raatz. Extending SLD-resolution to equational Horn clauses using E-unification. *Journal of Logic Programming*, 6:3–43, 1989.
- [115] J. H. Gallier and W. Snyder. A general complete E-unification procedure. In P. Lescanne, editor, *Rewriting Techniques and Applications*, volume 256 of *Lecture Notes in Computer Science*, pages 216–227. Springer-Verlag, 1987.
- [116] P. H. B. Gardiner, C. E. Martin, and O. De Moor. An algebraic construction of predicate transformers. *Science of Computer Programming*, 22(1-2):21–44, 1994.
- [117] D. Garlan and D.E. Perry. Introduction to the special issue on software architectures. *IEEE Transactions on Software Engineering*, 21:269–274, 1995.

- [118] M. Gelfond and V. Lifschitz. The Stable Model Semantics for Logic Programs. In R. A. Kowalski and K. A. Bowen, editors, *Proc. Fifth Int'l Conf. on Logic Programming*, pages 1070–1079. The MIT Press, 1988.
- [119] J. Y. Girard. Linear logic. *Theoretical Computer Science*, 50:1–102, 1987.
- [120] J.Y. Girard, Y. Lafont, and P. Taylor. *Proofs and Types*. Cambridge University Press, 1998.
- [121] J. Goguen and J. Meseguer. Eqlog: Equality, Types and Generic Modules for Logic Programming. In D. de Groot and G. Lindstrom, editors, *Logic Programming, Functions, Relations and Equations*, pages 295–363. Prentice-Hall, 1986.
- [122] S. Gregory. *Parallel Logic Programming in PARLOG: the Language and its Implementation*. Addison-Wesley, 1987.
- [123] John Guttag and James Horning. *Larch: languages and tools for formal specification*. Springer, 1993.
- [124] John Hannan and Dale Miller. From operational semantics to abstract machines. *Mathematical Structures in Computer Science*, 2(4):415–459, 1992. Invited to a special issue of papers selected from the 1990 Lisp and Functional Programming Conference.
- [125] J. Harland and D. Pym. A uniform proof-theoretic investigation of linear logic programming. *Journal of Logic and Computation*, 4(2), 1994.
- [126] J. Hennesy and D. Patterson. *Computer Architecture A Quantitative Approach*. Morgan Kaufman Publishers, Inc., 1990.
- [127] Joshua Hodas and Dale Miller. Representing objects in a logic programming language with scoping constructs. In David H. D. Warren and Peter Szeredi, editors, *1990 International Conference in Logic Programming*, pages 511–526. MIT Press, June 1990.
- [128] Joshua Hodas and Dale Miller. Logic programming in a fragment of intuitionistic linear logic. *Information and Computation*, 110(2):327–365, May 1994.
- [129] S. Hölldobler. Equational Logic Programming. In *Proc. Second IEEE Symp. on Logic In Computer Science*, pages 335–346. IEEE Computer Society Press, 1987.
- [130] K. Honda and M. Tokoro. An object calculus for asynchronous communication. In P. America, editor, *Proc. of ECOOP'91*, volume 512 of *Lecture Notes in Computer Science*, pages 133–147. Springer-Verlag, 1991.
- [131] P. Inverardi and A. L. Wolf. Formal specification and analysis of software architectures using the chemical abstract machine model. *IEEE Transactions on Software Engineering*, 21:373–386, 1995.
- [132] ISO/IEC JTC1 SC22 WG17 N.92. *PROLOG. Part 1. General Core, Committee Draft 1.0*, 1992.
- [133] J. Jaffar and J. L. Lassez. Constraint Logic Programming. In *Proc. Fourteenth Annual ACM Symp. on Principles of Programming Languages*, pages 111–119. ACM, 1987.
- [134] J. Jaffar and J. L. Lassez. Constraint Logic Programming. Technical report, Department of Computer Science, Monash University, June 1986.
- [135] J. Jaffar, J. L. Lassez, and J. W. Lloyd. Completeness of the Negation as Failure Rule. In *Proc. IJCAI-83*, pages 500–506, Karlsruhe, 1983.
- [136] J. Jaffar, J. L. Lassez, and M. J. Maher. A theory of complete logic programs with equality. *Journal of Logic Programming*, 3:211–223, 1984.
- [137] J. Jaffar and P. Stuckey. Canonical logic programs. *Journal of Logic Programming*, 3:143–155, 1986.
- [138] Joxan Jaffar and Jean-Louis Lassez. Constraint logic programming. In *Proceedings of Symposium on Principles of Programming Languages*. ACM, 1987.
- [139] Joxan Jaffar and Michael Maher. Constraint logic programming: A survey. *Journal of Logic Programming*, 19/20, 1994.

- [140] Bharat Jayaraman and Gopalan Nadathur. Implementation techniques for scoping constructs in logic programming. In Koichi Furukawa, editor, *Eighth International Logic Programming Conference*, pages 871–886. MIT Press, June 1991.
- [141] P.T. Johnstone. *Topos Theory*. Academic Press, 1977.
- [142] N. D. Jones and A. Mycroft. Stepwise Development of Operational and Denotational Semantics for PROLOG. In Sten-Åke Tärnlund, editor, *Proc. Second Int'l Conf. on Logic Programming*, pages 281–288, 1984.
- [143] N. D. Jones and H. Søndergaard. A Semantics-based Framework for the Abstract Interpretation of PROLOG. In S. Abramsky and C. Hankin, editors, *Abstract Interpretation of Declarative Languages*, pages 123–142. Ellis Horwood Ltd, 1987.
- [144] Neil D. Jones. Abstract interpretation and partial evaluation in functional and logic programming. In Maurice Bruynooghe, editor, *Logic Programming. Proceedings of the 1994 International Symposium*, pages 17–22. The MIT Press, 1994.
- [145] R. Kowalski. Algorithm = Logic + Control. *Communications of the ACM*, 22(7):424–436, 1979.
- [146] R. Kowalski. *Logic for Problem Solving*. North-Holland, 1979.
- [147] R. Kowalski. Problems and Promises of Computational Logic. In J. W. Lloyd, editor, *Computational Logic*. Springer-Verlag, 1990.
- [148] R. A. Kowalski. Predicate Logic as a Programming Language. In *Information Processing 74*, pages 569–574. North-Holland, 1974.
- [149] K. Kunen. Negation in logic programming. *Journal of Logic Programming*, 4:289–308, 1987.
- [150] Keehang Kwon. *Towards a Verified Abstract Machine for a Logic Programming Language with a Notion of Scope*. PhD thesis, Duke University, 1994. Available as Technical Report CS-1994-36, Department of Computer Science, Duke University.
- [151] Keehang Kwon. A structured presentation of a closure-based compilation method for a scoping notion in logic programming. *Journal of Universal Computer Science*, 3:4, 1997.
- [152] Keehang Kwon, Gopalan Nadathur, and Debra Sue Wilson. Implementing a notion of modules in the logic programming language  $\lambda$ Prolog. In Evelina Lamma and Paola Mello, editors, *Extensions of Logic Programming: Proceedings of the Third International Workshop, Bologna, Italy*, volume 660 of *Lecture Notes in Artificial Intelligence*, pages 359–393. Springer-Verlag, 1993.
- [153] Keehang Kwon, Gopalan Nadathur, and Debra Sue Wilson. Implementing polymorphic typing in a logic programming language. *Computer Languages*, 20(1):25–42, 1994.
- [154] T. K. Lakshman and Uday S. Reddy. Typed Prolog: A semantic reconstruction of the Mycroft-O'Keefe type system. In *Proceedings of the 1991 IEEE Symposium on Logic Programming*, pages 202–220, 1991.
- [155] J. L. Lassez and K. MacAloon. A constraint sequent calculus. In *Proceedings of the 5th Symp. on LICS*, 1990.
- [156] F.W. Lawvere. Functorial semantics of algebraic theories. *Proceedings of the National Academy of Sciences*, 50:869–872, 1963.
- [157] James Lipton and Paul Broome. Combinatory logic programming. In *Proc. ILPS'94*. MIT, 1994.
- [158] James Lipton and Michael O'Donnell. Some intuitions behind realizability semantics for constructive logic: Tableaux and läuchli countermodels. *Annals of Pure and Applied Logic*, 81:87–111, 1996.
- [159] J. W. Lloyd. *Foundations of Logic Programming*. Springer-Verlag, 1987. Second edition.
- [160] J. W. Lloyd and J. C. Shepherdson. Partial Evaluation in Logic Programming. *Journal of Logic Programming*, 11:217–242, 1991.
- [161] J. Lobo, J. Minker, and A. Rajasekar. *Foundations of Disjunctive Logic Programming*. The MIT Press, 1992.



- [162] Donald W. Loveland and Gopalan Nadathur. Proof procedures for logic programming. Technical Report CS-1994-19, Department of Computer Science, Duke University, May 1994. To appear in *Handbook of Logic in Artificial Intelligence and Logic Programming*, D. Gabbay, C. Hogger and A. Robinson (eds.), Oxford University Press.
- [163] Donald W. Loveland and Gopalan Nadathur. Proof procedures for logic programming. In *Handbook of Logic in Artificial Intelligence and Logic Programming*, volume 5, pages 163–234. Oxford University Press, 1998.
- [164] G. Levi M. Falaschi and C. Palamidessi. A synchronization logic: Axiomatics and formal semantics of generalized horn clauses. *Information and Control*, 60(1-3):36–69, Jan/Feb/Mar 1984.
- [165] J. Magee, N. Dulay, S. Eisenbach, and J. Kramer. Specifying distributed software architectures. In *Proceedings of the 5th European Software Engineering Conference*, 1995.
- [166] P. Mancarella, S. Martini, and D. Pedreschi. Complete Logic Programs with Domain Closure Axiom. *Journal of Logic Programming*, 5(3):263–276, 1988.
- [167] K. Marriott and H. Søndergaard. Bottom-up Abstract Interpretation of Logic Programs. In R. A. Kowalski and K. A. Bowen, editors, *Proc. Fifth Int'l Conf. on Logic Programming*, pages 733–748. The MIT Press, 1988.
- [168] K. Marriott and H. Søndergaard. Abstract Interpretation of Logic Programs: the Denotational Approach. In A. Bossi, editor, *Proc. Fifth Italian Conference on Logic Programming*, pages 399–425, 1990.
- [169] K. Marriott and H. Søndergaard. Analysis of Constraint Logic Programs. In S. K. Debray and M. Hermenegildo, editors, *Proc. North American Conf. on Logic Programming '90*, pages 531–547. The MIT Press, 1990.
- [170] P. Martin-Lof. Constructive mathematics and computer programming. *Logic, Methodology and Philosophy of Science*, 6:153–175, 1980.
- [171] Raymond McDowell and Dale Miller. A logic for reasoning with higher-order abstract syntax. In Glynn Winskel, editor, *Proceedings, Twelfth Annual IEEE Symposium on Logic in Computer Science*, Warsaw, Poland, July 1997. IEEE Computer Society Press.
- [172] R. McGrail. *Monads and Control in Logic Programming*. PhD thesis, Wesleyan University, 1997. To appear.
- [173] Robert McGrail and Jim Lipton. Encapsulating data in logic programming via categorical constraints. In *Principles of Declarative Programming, proceedings of ALP'98 and PLILP '98*, volume LNCS 1490, pages 391–410. Springer, 1998.
- [174] Maria Chiara Meo. *A Framework for Reasoning about the Semantics of Logic Programs*. PhD thesis, University of Pisa, 1996.
- [175] J. Meseguer. Multiparadigm logic programming. In H. Kirchner and G. Levi, editors, *Algebraic and Logic Programming, Proceedings of the Third International Conference*, volume 632 of *Lecture Notes in Computer Science*, pages 158–200. Springer-Verlag, 1992.
- [176] D. Miller. A Logical Analysis of Modules in Logic Programming. *Journal of Logic Programming*, 6:79–108, 1989.
- [177] D. Miller. The  $\pi$ -calculus as a theory in linear logic: Preliminary result. In E. Lamma and P. Mello, editors, *Proc. of Third International Workshop on Extensions of Logic Programming*, volume 660 of *Lecture Notes in Computer Science*, pages 242–264. Springer-Verlag, 1992.
- [178] D. Miller, F. Pfenning, G. Nadathur, and A. Scedrov. Uniform proofs as a foundation for Logic Programming. *Annals of Pure and Applied Logic*, 51:125–157, 1991.
- [179] Dale Miller. A theory of modules for logic programming. In Robert M. Keller, editor, *Third Annual IEEE Symposium on Logic Programming*, pages 106–114, Salt Lake City, Utah, September 1986.
- [180] Dale Miller. A logical analysis of modules in logic programming. *Journal of Logic Programming*, 6(1-2):79–108, January 1989.

- [181] Dale Miller. A logic programming language with lambda-abstraction, function variables, and simple unification. *Journal of Logic and Computation*, 1(4):497–536, 1991.
- [182] Dale Miller. Abstract syntax and logic programming. In *Logic Programming: Proceedings of the First and Second Russian Conferences on Logic Programming*, number 592 in Lecture Notes in Artificial Intelligence, pages 322–337. Springer-Verlag, 1992.
- [183] Dale Miller. The  $\pi$ -calculus as a theory in linear logic: Preliminary results. In E. Lamma and P. Mello, editors, *Proceedings of the 1992 Workshop on Extensions to Logic Programming*, number 660 in LNCS, pages 242–265. Springer-Verlag, 1993.
- [184] Dale Miller. A multiple-conclusion meta-logic. In S. Abramsky, editor, *Ninth Annual Symposium on Logic in Computer Science*, pages 272–281, Paris, July 1994.
- [185] Dale Miller. Forum: A multiple-conclusion specification language. *Theoretical Computer Science*, 165(1):201–232, September 1996.
- [186] Dale Miller and Gopalan Nadathur. A logic programming approach to manipulating formulas and programs. In Seif Haridi, editor, *IEEE Symposium on Logic Programming*, pages 379–388, San Francisco, September 1987.
- [187] Dale Miller and Gopalan Nadathur.  $\lambda$ Prolog version 2.7. Distributed in C-Prolog and Quintus Prolog source code, August 1988.
- [188] Dale Miller, Gopalan Nadathur, Frank Pfenning, and Andre Scedrov. Uniform proofs as a foundation for logic programming. *Annals of Pure and Applied Logic*, 51:125–157, 1991.
- [189] Dale Miller, Gopalan Nadathur, Frank Pfenning, and Andre Scedrov. Uniform proofs as a foundation for logic programming. *Annals of Pure and Applied Logic*, 51:125–157, 1991.
- [190] Robin Milner, Joachim Parrow, and David Walker. A calculus of mobile processes, Part I. *Information and Computation*, pages 1–40, September 1992.
- [191] Robin Milner, Joachim Parrow, and David Walker. A calculus of mobile processes, Part II. *Information and Computation*, pages 41–77, September 1992.
- [192] Robin Milner, Mads Tofte, and Robert Harper. *The Definition of Standard ML*. MIT Press, 1990.
- [193] J. Minker and A. Rajasekar. A Fixpoint Semantics for Disjunctive Logic Programs. *Journal of Logic Programming*, 9:45–74, 1990.
- [194] P. Mishra. Towards a Theory of Types in Prolog. In *Proc. 1st IEEE Symp. Logic Programming*, pages 289–298, 1984.
- [195] E. Moggi. Notions of computation and monads. *Information and Computation*, 93, 1991.
- [196] J. J. Moreno and M. Rodriguez-Artalejo. Logic Programming with Functions and Predicates: The language Babel. *Journal of Logic Programming*, 1991.
- [197] A. Mycroft and R. A. O’Keefe. A Polymorphic Type System for Prolog. *Artificial Intelligence*, 23(3):295–307, 1984.
- [198] Gopalan Nadathur. *A Higher-Order Logic as the Basis for Logic Programming*. PhD thesis, University of Pennsylvania, 1987.
- [199] Gopalan Nadathur. A proof procedure for the logic of hereditary Harrop formulas. *Journal of Automated Reasoning*, 11(1):115–145, August 1993.
- [200] Gopalan Nadathur. A proof procedure for the logic of hereditary Harrop formulas. *Journal of Automated Reasoning*, 11(1):115–145, August 1993.
- [201] Gopalan Nadathur. A fine-grained notation for lambda terms and its use in intensional operations. Technical Report TR-96-13, Department of Computer Science, University of Chicago, May 1996. To appear in *Journal of Functional and Logic Programming*.
- [202] Gopalan Nadathur. Uniform provability in classical logic. Technical Report TR-96-09, Department of Computer Science, University of Chicago, March 1996. To appear in *Journal of Logic and Computation*.

- [203] Gopalan Nadathur. Correspondences between classical, intuitionistic and uniform provability. Technical Report TR-97-12, Department of Computer Science, University of Chicago, November 1997. To appear in *Theoretical Computer Science*.
- [204] Gopalan Nadathur and Bharat Jayaraman. Towards a WAM model for  $\lambda$ Prolog. In Ewing Lusk and Ross Overbeek, editors, *Proceedings of the North American Conference on Logic Programming*, pages 1180–1198, Cleveland, Ohio, October 1989. MIT Press.
- [205] Gopalan Nadathur, Bharat Jayaraman, and Keehang Kwon. Scoping constructs in logic programming: Implementation problems and their solution. *Journal of Logic Programming*, 25(2):119–161, November 1995.
- [206] Gopalan Nadathur, Bharat Jayaraman, and Debra Sue Wilson. Implementation considerations for higher-order features in logic programming. Technical Report CS-1993-16, Department of Computer Science, Duke University, June 1993.
- [207] Gopalan Nadathur and Donald W. Loveland. Uniform proofs and disjunctive logic programming. In Dexter Kozen, editor, *Tenth Annual IEEE Symposium on Logic in Computer Science*, pages 148–155. IEEE Computer Society Press, June 1995.
- [208] Gopalan Nadathur and Dale Miller. An Overview of  $\lambda$ Prolog. In *Fifth International Logic Programming Conference*, pages 810–827, Seattle, Washington, August 1988. MIT Press.
- [209] Gopalan Nadathur and Dale Miller. An overview of  $\lambda$ Prolog. In Kenneth A. Bowen and Robert A. Kowalski, editors, *Fifth International Logic Programming Conference*, pages 810–827, Seattle, Washington, August 1988. MIT Press.
- [210] Gopalan Nadathur and Dale Miller. Higher-order horn clauses. *Journal of the ACM*, 37(4):777–814, 1990.
- [211] Gopalan Nadathur and Dale Miller. Higher-order Horn clauses. *Journal of the ACM*, 37(4):777–814, October 1990.
- [212] Gopalan Nadathur and Dale Miller. Higher-order logic programming. In C. Hogger D. Gabbay and A. Robinson, editors, *Handbook of Logic in Artificial Intelligence and Logic Programming*, volume 5, pages 499–590. Oxford University Press, 1998.
- [213] Gopalan Nadathur, Dustin Mitchell, Shyan-Ming Perng, and Guanshan Tong. Chicago Lambda Prolog, October 1998. A compiler and abstract machine based implementation of  $\lambda$ Prolog (beta release).
- [214] Gopalan Nadathur and Frank Pfenning. The type system of a higher-order logic programming language. In Frank Pfenning, editor, *Types in Logic Programming*, pages 245–283. MIT Press, 1992.
- [215] Gopalan Nadathur and Frank Pfenning. The type system of a higher-order logic programming language. In Frank Pfenning, editor, *Types in Logic Programming*, pages 245–283. MIT Press, 1992.
- [216] Gopalan Nadathur and Guanshan Tong. Realizing modularity in  $\lambda$ Prolog. Technical Report TR-97-07, Department of Computer Science, University of Chicago, August 1997.
- [217] Gopalan Nadathur and Debra Sue Wilson. A representation of lambda terms suitable for operations on their intensions. In *Proceedings of the 1990 ACM Conference on Lisp and Functional Programming*, pages 341–348. ACM Press, 1990.
- [218] Gopalan Nadathur and Debra Sue Wilson. A notation for lambda terms: A generalization of environments. Technical Report CS-1997-01, Department of Computer Science, University of Chicago, January 1997. To appear in *Theoretical Computer Science*.
- [219] Gopalan Nadathur and Debra Sue Wilson. A notation for lambda terms: A generalization of environments. *Theoretical Computer Science*, 198(1-2):49–98, 1998.
- [220] L. Naish. *Negation and Control in Prolog*, volume 238 of *Lecture Notes in Computer Science*. Springer-Verlag, 1986.
- [221] A. Nerode, J. Remmel, and W. Marek. A theory of nonmonotonic rule systems i. *Annals of Mathematics and Artificial Intelligence*, 1, 1990.

- [222] Uwe Nestmann and Benjamin C. Pierce. Decoding choice encodings. In *Proc. of CONCUR'96*, volume 1119 of *Lecture Notes in Computer Science*, pages 179–194. Springer-Verlag, 1996.
- [223] P. W. O'Hearn and R. D. Tennent. Parametricity and local variables. Technical report, Syracuse University, 1993.
- [224] L. Ong. Non-determinism in functional programming. In *Proceedings of the 8th IEEE Symposium on Logic and Computer Science*, 1993.
- [225] Friedrich Otto, Paliath Narendran, and Daniel J. Dougherty. Some independence results for equational unification. In *Proc. Rewriting Techniques and Applications*, volume 914, pages 367–381, 1995.
- [226] Friedrich Otto, Paliath Narendran, and Daniel J. Dougherty. Some independence results for equational unification. *Theoretical Computer Science*, 198:1–47, 1998.
- [227] Catuscia Palamidessi. Comparing the expressive power of the synchronous and the asynchronous  $\pi$ -calculus. In *Proceedings of the 24th ACM Symposium on Principles of Programming Languages*, pages 256–265. ACM, 1997.
- [228] Catuscia Palamidessi. Constraint programming for concurrent and distributed computing. In Hermès, editor, *Proc. of the JFPLC-UNIF '97*, pages 11–20, 1997. Invited talk.
- [229] Catuscia Palamidessi and Joachim Parrow, editors. *Proceedings of EXPRESS'97*, volume 7 of *Electronic Notes in Theoretical Computer Science*. Elsevier Science B.V., 1997. Available at the web address <http://www.elsevier.nl/locate/entcs/volume7.html>.
- [230] Catuscia Palamidessi and Joachim Parrow, editors. *Special Issue dedicated to Express'97*, Information and Computation, Forthcoming.
- [231] P. Panangaden, V. Saraswat, P.J. Scott, and R.A.G. Seely. A hyperdoctrinal view of constraint systems. In *Lecture Notes in Computer Science 666*. Springer Verlag, 1993.
- [232] John Penix and Perry Alexander. Declarative specification of software architectures. 1996.
- [233] John Penix and Perry Alexander. Toward automated component adaptation. In *Proceedings of the 9th Int. Conference on Software Engineering and Knowledge Engineering*, 1997.
- [234] D.E. Perry and A.L. Wolf. Foundations for the study of software architectures. In *ACM SIGSOFT Notes*, volume 17,4, pages 40–52, 1992.
- [235] Frank Pfenning. *Types in Logic Programming*, chapter Dependent Types in Logic Programming. MIT, 1992.
- [236] David Pitt, Samson Abramsky, Axel Poigne, and David Rydeheard, editors. *Category Theory and Computer Programming*. Springer-Verlag, New York, 1985. Lecture Notes in Computer Science 240.
- [237] G. D. Plotkin. Building-in equational theories. *Machine Intelligence*, 7:73–90, 1972.
- [238] A. Poigné. Algebra categorically. In *Category Theory and Computer Programming*. Springer, 1986.
- [239] A. J. Power and C. Sterling. A notion of map between logic programs. In D.H.D. Warren and P. Szeredi, editors, *Proc. Seventh International Symposium of Logic Programming*. The MIT Press, 1990.
- [240] J. Power and Y. Kinoshita. A new foundation for logic programming. In *Extensions of Logic Programming '96*. Springer Verlag, 1996.
- [241] T. Przymusiński. On constructive negation in logic programming. In *Proc. North American Conference on Logic Programming*, 1989.
- [242] D. Pym. Functorial kripke models of the  $\lambda\pi$ -calculus. Lecture at Newton Institute Semantics Programme, Workshop on Category Theory and Logic Programming, Cambridge, September 1995, 1995.
- [243] Y.S. Ramakrishna, C.R. Ramakrishnan, I.V. Ramakrishnan, Scott Smolka, Terry Swift, and David S. Warren. Efficient model checking using tabled resolution. <http://www.cs.sunysb.edu/cram/papers/model/xmc.html>, 1997. (To be presented at CAV'97).

- [244] U. S. Reddy. Transformation of Logic Programs into Functional Programs. In *Proc. of Symposium on Logic Programming - Atlantic City*, pages 188–196. IEEE Computer Society, 1984.
- [245] U. S. Reddy. Narrowing as the Operational Semantics of Functional Languages. In *Proc. Second IEEE Int'l Symp. on Logic Programming*, pages 138–151. IEEE, 1985.
- [246] U. S. Reddy. Notions of polymorphism for predicate logic programs. In *Proceedings of the 5th International Conference on Logic Programming, Seattle*, 1988.
- [247] M.D. Rice and S.B Seidman. A formal model for module interconnection languages. *IEEE Transactions on Software Engineering*, 20(1):88–101, January 1994.
- [248] M.D. Rice and S.B Seidman. Using z as a substrate for an architectural style description language. Technical Report Technical Report CS-96-120, Colorado State University, Fort Collins CO, 1996.
- [249] David Walker Robin Milner, Joachim Parrow. Modal logics for mobile processes. *Theoretical Computer Science*, 114(1):149–171, 1993.
- [250] J. A. Robinson. A machine-oriented logic based on the resolution principle. *Journal of the ACM*, 12(1):23–41, January 1965.
- [251] J. A. Robinson. Beyond LOGLISP: combining functional and relational programming in a reduction setting. In D. Michie and Hayes-Michie, editors, *Machine Intelligence 11*. Oxford University Press, 1985.
- [252] Matthew Ruhlen. Development of a rewrite system for relational algebra equations modeling horn clause resolution. Master's thesis, Wesleyan University, 1997.
- [253] D.E. Rydeheard and R.M. Burstall. A categorical unification algorithm. In *Category Theory and Computer Programming*, 1985.
- [254] V. A. Saraswat. *Concurrent Constraint Programming Languages*. PhD thesis, Carnegie-Mellon University, January 1989.
- [255] V. A. Saraswat, M. Rinard, and P. Panangaden. Semantic Foundation of Concurrent Constraint Programming. In *Proc. Eighteenth Annual ACM Symp. on Principles of Programming Languages*, pages 333–353. ACM, 1991.
- [256] Vijay A. Saraswat, Martin Rinard, and Prakash Panangaden. Semantic foundations of concurrent constraint programming. In *Conf. Record 18th Annual ACM Symp. on Principles of Programming Languages, Orlando, FL, USA, 1991*, pages 333–52. 1990.
- [257] J. S. Schlipf. The expressive powers of the logic programming semantics. In *Ninth ACM Symposium on Principles of Database Systems*, pages 196–204. ACM Press, 1990.
- [258] J. S. Schlipf. Formalizing a logic for logic programming. *Annals of Mathematics and Artificial Intelligence*, 5:279–302, 1992.
- [259] R. A. G. Seely. Hyperdoctrines, natural deduction and the Beck condition. *ZML*, 29, 1983.
- [260] E. Y. Shapiro. The family of concurrent logic programming languages. *ACM Computing Surveys*, 21(3):412–510, 1989.
- [261] M. Shaw and D. Garlan. *Software Architecture: Perspectives on an Emerging Discipline*. Prentice-Hall, 1996.
- [262] J. C. Shepherdson. Negation as Failure II. *Journal of Logic Programming*, 2(3):185–202, 1985.
- [263] J. C. Shepherdson. Negation in logic programming. In J. Minker, editor, *Foundations of Deductive Databases and Logic Programming*, pages 19–88. Morgan Kaufmann, 1988.
- [264] J. C. Shepherdson. Language and equality theory in logic programming. Technical Report PM-91-02, School of Mathematics, University of Bristol, 1991.
- [265] J. H. Siekmann. Unification Theory. *Journal of Symbolic Computation*, 7:207–274, 1989.
- [266] Alex K. Simpson. Compositionality via cut-elimination: Hennessy-milner logic for an arbitrary gsos. In D. Kozen, editor, *Proceedings of LICS'95*, pages 420–430, San Diego, California, jun 1995. IEEE Computer Society Press.

- [267] Douglas Smith. Kids: A semiautomatic program development system. In *IEEE Transactions on Software Engineering*, volume 16(9), pages 1024–1043, 1990.
- [268] G. Smolka. *Logic Programming over Polymorphically Order-Sorted Types*. PhD thesis, Universität Kaiserslautern, 1989.
- [269] Gert Smolka. A foundation for higher-order concurrent constraint programming. In J.-P. Jouannaud, editor, *Proceedings of Constraints in Computational Logics*, volume 845 of *lncs*, pages 50–72. sv, 1994. Available as Research Report RR-94-16 from DFKI Kaiserslautern.
- [270] M. Spivey. A functional theory of exceptions. *Science of Computer Programming*, 14(1):25–42, 1990.
- [271] T. Streicher. *Semantics of Type Theory*. Birkhäuser, 1992.
- [272] P. Stuckey. Constructive negation for constraint logic programming. In *Proceedings of the 6th IEEE Symposium on Logic and Computer Science*, 1991.
- [273] P. J. Stuckey. Negation and constraint logic programming. *Information and Computation*, 118, 1995.
- [274] V. S. Subrahmanian. A simple formulation of the theory of metalogic programming. In H. Abramson and M. H. Rogers, editors, *Meta-programming in Logic Programming*, pages 65–101. The MIT Press, 1989.
- [275] Alfred Tarski and Steven Givant. *A formalization of Set Theory without variables*. American Mathematical Society, Providence, R.I., 1987.
- [276] A. S. Troelstra and D. van Dalen. *Constructivism in Mathematics: An Introduction*, volume 2. Elsevier Science Publishers, 1988.
- [277] M. H. van Emden and R. A. Kowalski. The semantics of predicate logic as a programming language. *Journal of the ACM*, 23(4):733–742, 1976.
- [278] A. van Gelder, K. A. Ross, and J. S. Schlipf. The Well-Founded Semantics for General Logic Programs. *Journal of the ACM*, 38(3):620–650, 1991.
- [279] P. Wadler. Comprehending monads. *Mathematical Structures in Computer Science*, 2, 1992.
- [280] P. Wadler. How to declare an imperative. In *Proc. International Logic Programming Symposium*. MIT Press, 1995.
- [281] D. Warren and M. Kifer. Hilog : A Foundation for Higher-Order Logic Programming. In E. Lusk and R. Overbeek, editors, *Proc. North American Conf. on Logic Programming'89*, pages 1090–1114. The MIT Press, 1989.
- [282] D. H. Warren. An Abstract Prolog Instruction Set. Technical Report Note 309, SRI International, 1983.
- [283] D. S. Warren, M. Hermenegildo, and S. K. Debray. On the practicality of global flow analysis of logic programs. In R. A. Kowalski and K. A. Bowen, editors, *Proc. Fifth Int'l Conf. on Logic Programming*, pages 684–699, 1988.
- [284] D.A. Wolfram. A semantics for  $\lambda$ Prolog. *Theoretical Computer Science*, 136(1):277–289, 1994.
- [285] P. Zave and M. Jackson. Conjunction as composition. *ACM Transactions on Software Engineering*, 2(4):379–411, October 1993.