Inductive types in constructive languages

Bruin, Peter Johan de

Bibliography

[1] C. J. Aarts, R. C. Backhouse, P. Hoogendijk, T. S. Voermans, and J. van der Woude, *A Relational Theory of Datatypes*. Available via anonymous ftp from `ftp.win.tue.nl` in directory `pub/math.prog.construction`, Eindhoven University of Technology 1992. 12

[2] H. Abrahamson and V. Dahl, *Logic Grammars*. Springer-Verlag 1989. 19

[3] Peter Aczel, *An introduction to Inductive Definitions*. In: **Handbook of Mathematical Logic** (ed. Jon Barwise), North Holland 1977, pp. 739–782. 44, 45, 47

[4] Peter Aczel, *Non-well-founded Sets*. CSLI Lecture Notes no. 14, Stanford 1988. 93, 125

[5] Roland Backhouse, Paul Chisholm, Grant Malcolm, and Erik Saaman, *Do-it-yourself type theory*. **Formal Aspects of Computing 1** (1989), pp. 19–84.

[6] R.C. Backhouse, *On a Relation on Functions*. In: **Beauty Is Our Business—A Birthday Salute to Edsger W. Dijkstra** (ed. W.H.J. Feijen e.a.), Springer Verlag 1990, pp. 7–18. 153

[7] R.C. Backhouse, P.J. de Bruin, G.R. Malcolm, T.S. Voermans, and J.C.S.P. van der Woude, *Relational catamorphisms*. In: **Constructing Programs From Specifications**, North Holland 1991, pp. 287–318.

[8] Roland Backhouse and Henk Doornbos, *Mathematical Induction Made Calculational*. CS-report 94-16, Eindhoven University of Technology 1994. 12

[9] E.S. Bainbridge, P.J. Freyd, A. Scedrov, and P.J. Scott, *Functorial Polymorphism*. **Theoretical Computer Science 70** (1990), pp. 35–64. 145

[10] Erik Barendsen and Marc Bezem, *Bar Recursion versus Polymorphism*. Technical Report 81, Utrecht Research Institute for Philosophy, Utrecht University 1991. 112

[11] N.G. de Bruijn, *A survey of the project AUTOMATH*. In: **To H.B. Curry: Essays on Combinatory Logic, Lambda Calculus and Formalism** (ed. Seldin and Hindley), Academic Press 1980, pp. 579–607. 11

[12] N.G. de Bruijn, *The Mathematical Vernacular, a language for mathematics with typed sets*. In: **Workshop on Programming Logic**, Marstrand Sweden 1987. 11

[13] P.J. de Bruin, *Towards decidable Constructive Type Theories as practical descriptive and programming languages*. Master's thesis, report 87-6, Dept. of Informatics, University of Nijmegen 1987.

[14] P.J. de Bruin, *Naturalness of Polymorphism*. Report CS8916, Dept. of Mathematics and Computing Science, University of Groningen 1989. 144

[15] P.J. de Bruin, *Proof elimination in Type Theory*. Report CS9202, Dept. of Mathematics and Computing Science, University of Groningen 1992. 135

[16] R. Burstall and B. Lampson, *A kernel language for abstract data types and modules.* In: **Semantics of Data Types 1984**, LNCS 173, pp. 1–50. 30

[17] A. Carboni, P.J. Freyd, and A. Scedrov, *A Categorical Approach to Realizability and Polymorphic Types.* In: **Mathematical Foundations of Programming Language Semantics 1987**, LNCS 298, pp. 23–42. 145

[18] R.L. Constable e.a., *Implementing Mathematics with the Nuprl Proof Development System.* Prentice Hall 1986. 11, 72, 102, 139

[19] Robert L. Constable, *Type Theory as a Foundation for Computer Science.* In: **Theoretical Aspects of Computer Science 1991**, LNCS 526, pp. 226–243. 137

[20] R.L. Constable and N.P. Mendler, *Recursive Definitions in Type Theory.* In: **Logics of Programs 1985**, LNCS 193, pp. 61–78.

[21] Th. Coquand and G. Huet, *A Theory of Constructions.* In: **Semantics of Data Types** (ed. G. Kahn e.a.), Sophia Antipolis 1985. 12, 15, 107, 135

[22] Thierry Coquand and Christine Paulin, *Inductively defined types.* In: **COLOG-88**, LNCS 417, pp. 50–66, and **Workshop on Programming Logic 1989**, report 54, Programming Methodology Group, Göteborg, pp. 191–208. 72, 78, 108

[23] Thierry Coquand, *Pattern matching with dependent types.* In: **Proceedings of the 1992 Workshop on Types for Proofs and Programs**, Göteborg 1992. 12

[24] D. DeGroot and G. Lindstrom (ed.), *Logic Programming: Functions, Relations, and Equations*, Prentice Hall 1986. 115, 161

[25] Peter Dybjer and Herbert Sander, *A Functional Programming Approach to the Specification and Verification of Concurrent Systems.* In: **Workshop on Specification and Verification of Concurrent Systems**, Stirling 1988, and **Formal Aspects of Computing 1** (1989), pp. 303–319. 88

[26] Peter Dybjer, *An inversion principle for Martin-Löf's type theory.* In: **Workshop on Programming Logic 1989**, report 54, Programming Methodology Group, Göteborg, pp. 177–190.

[27] Roy Dyckhoff, *Category Theory as an extension of Martin-Löf Type Theory.* Report CS/85/3, Dept. of Computational Science, University of St. Andrews 1985.

[28] H.-D. Ehrich, *Specifying algebraic data types by domain equations.* In: **Foundations of Computation Theory 1981**, LNCS 117, pp. 120–129.

[29] Maarten M. Fokkinga and Erik Meijer, *Program Calculation Properties of Continuous Algebras.* Report CS-R9104, CWI Amsterdam 1991.

[30] Maarten M. Fokkinga, *Law and Order in Algorithmics.* Ph.D. Thesis, Twente University of Technology 1992. 52, 57, 59, 75

[31] G. Frege, *Grundgesetze der Arithmetik* (vol. 1), Jena 1893. 136

[32] P.J. Freyd, J.Y. Girard, A. Scedrov, and P.J. Scott, *Semantic Parametricity in Polymorphic Lambda Calculus.* In: **Logic in Computing Science 1988**, IEEE, pp. 274–279. 145

[33] A.J.M. van Gasteren, *On the shape of mathematical arguments.* Ph.D. thesis, Eindhoven 1988, and LNCS 445 (1990).

[34] J.A. Goguen and J. Meseguer, *Eqlog: equality, types, and generic modules for logic programming.* In [24], pp. 295–363.

[35] M. Gordon, R. Milner, and C. Wadsworth, *Edinburgh LCF*. LNCS 78 (1979).   11, 101

[36] John W. Gray, *A Categorical Treatment of Polymorphic Operations*. In: **Mathematical Foundations of Programming Language Semantics 1987**, LNCS 298, pp. 2–22.   145

[37] Tatsuya Hagino, *A Typed Lambda Calculus with Categorical Type Constructors*. In: **Category Theory and Computer Science 1987**, LNCS 283, pp. 140–157.   50, 54, 146

[38] P. Hudak and P. Wadler, editors, *Report on the Functional Programming Language Haskell*. Technical Report, Yale University and University of Glasgow, Dept. of Computer Science, December 1988.   155

[39] Martin C. Henson and Raymond Turner, *A Constructive Set Theory for Program Development*. In: **8th Conf. on Foundations of Software Technology and Theoretical Computer Science**, LNCS 338 (1988), pp. 329–347.   109, 110

[40] Martin C. Henson, *Program Development in the Constructive Set Theory TK*. **Formal Aspects of Computing 1** (1989), pp. 173–192.   110

[41] C.A.R. Hoare, *Communicating Sequential Processes*. **Communications of the ACM 21** (1978), pp. 666–677.   88

[42] G. Huet and G. Plotkin (eds.), *Logical Frameworks*. Cambridge 1991.   116

[43] Bart Jacobs, *The Inconsistency of Higher Order Extensions of Martin-Löf's Type Theory*. **Journ. Philosophical Logic 18** (1988), pp. 399–422.   136

[44] Bart Jacobs, *Categorical Type Theory*. Ph.D. Thesis, University of Nijmegen, 1991.   10

[45] Robert Kerkhoff, *Eine Konstruktion absolut freier Algebren*. **Mathematische Annalen 158** (1969), pp. 109–112.   92, 94, 142

[46] J. Lambek and P.J. Scott, *Introduction to higher order categorical logic*. Cambridge 1986.   13, 61, 115

[47] Leslie Lamport, *How to Write a Proof*. SRC report 94, DEC Systems Research Center 1993.   116

[48] Zhaohui Luo, *ECC, an Extended Calculus of Constructions*. In: **Logic in Computer Science 1989**, IEEE, pp. 386–395.   107, 126, 136

[49] QingMing Ma and John C. Reynolds, *Types, Abstraction, and Parametric Polymorphism, Part 2*. In: **Mathematical Foundations of Programming Semantics 1991**, LNCS 598, pp. 1–40.

[50] Lena Magnuson and Bengt Nordström, *The ALF Proof Editor and its Proof Engine*. In: **Types for Proofs and Programs** (Nijmegen 1993), LNCS 806, pp. 213–237.   12, 114

[51] S. Mac Lane, *Categories for the working mathematician*. Graduate Texts in Mathematics 5, Springer-Verlag 1971.   145, 153

[52] Grant Malcolm, *Algebraic Data Types and Program Transformation*. Ph.D. Thesis, University of Groningen 1990.   42, 75, 89, 152

[53] Jan Małuszyński, *Attribute Grammars and Logic Programs: A Comparison of Concepts*. In: **Attribute Grammars, Applications and Systems**, Prague 1991, LNCS 545, pp. 330–357.   19

[54] Ernest G. Manes, *Algebraic Theories*. Graduate Texts in Mathematics 26, Springer-Verlag 1976.   57, 61, 65, 92, 120

[55] P. Martin-Löf, *Hauptsatz for the Intuitionistic Theory of Iterated Inductive Definitions.* In: **Second Scandinavian Logic Symposium** (ed. J.E. Fenstad), North-Holland 1971, pp. 179–216. 42, 45

[56] P. Martin-Löf, *Constructive Mathematics and Computer Programming.* In: **Logic, Methodology, and Philosophy of Science VI**, 1979 (ed. L.J. Cohen e.a.), North-Holland 1982, pp. 153–175. 11, 14, 68, 135

[57] Lambert Meertens, *Constructing a calculus of programs.* In: **Mathematics of Program Construction 1989** (ed. J.L.A. van de Snepscheut), LNCS 375, pp. 66–90. 54, 74

[58] L.G.L.T. Meertens, *Paramorphisms.* **Formal Aspects of Computing 4** (1992), pp. 413–424. 74

[59] N.P. Mendler, *Inductive Definition in Type Theory.* Ph.D. Thesis, Cornell University 1987. 14, 78

[60] N.P. Mendler, *Recursive Types and Type Constraints in Second-Order Lambda Calculus.* In: **Logic in Computer Science 1987**, IEEE, pp. 30–36. 79

[61] N.P. Mendler, *Predicative Type Universes and Primitive Recursion.* In: **Logic in Computer Science 1991**, IEEE, pp. 173–185. 110

[62] J.C. Mitchell and A.R. Meyer, *Second-order logical relations.* In: **Logics of Programs 1985**, LNCS 193, pp. 225–236. 145, 148

[63] J.D. Monk, *Introduction to Set Theory.* McGraw-Hill 1969. 120, 122

[64] Yiannis N. Moschovakis, *Elementary Induction on Abstract Structures.* Studies in Logic and the Foundation of Mathematics, North Holland 1974.

[65] P. Odifreddi, *Classical Recursion Theory.* Studies in Logic and the Foundation of Mathematics, North Holland 1989. 14

[66] Christian-Emil Ore, *The Extended Calculus of Constructions (ECC) with Inductive Types.* **Information and Computation 99** (1992), pp. 231-264. 107, 108

[67] Ross Paterson, *Reasoning about Functional Programs.* Ph.D. thesis, University of Queensland 1987. 100

[68] Christine Paulin-Mohring, *Inductive Definitions in the system Coq — Rules and Properties.* In: **Typed Lambda Calculi and Applications** (Utrecht 1993), LNCS 664, pp. 328–345. 12, 72

[69] Lawrence C. Paulson, *Constructing Recursion Operators in Intuitionistic Type Theory.* Cambridge 1984. 47

[70] Lawrence C. Paulson, *Logic and computation — Interactive proof with Cambridge LCF.* Cambridge 1987. 11, 101

[71] Duško Pavlović, *Constructions and Predicates.* In: **Category Theory and Computer Science 1991**, LNCS 530, pp. 173–196. 137

[72] Kent Petersson and Dan Synek, *A Set Constructor for Inductive Sets in Martin-Löf's Type Theory.* In: **Workshop on Programming Logic 1989**, report 54, Programming Methodoly Group, Göteborg, pp. 162–175. 69

[73] F. Pfenning and Ch. Paulin-Mohring, *Inductively Defined Types in the Calculus of Constructions.* In: **Mathematical Foundations of Programming Semantics 1989**, LNCS 442, pp. 209–228. 12, 108

[74] G.D. Plotkin, *Lambda-definability in the full type hierarchy*. In: **To H.B. Curry: Essays on combinatory logic, lambda calculus, and formalism** (ed. Seldin and Hindley), Academic Press, New York 1980, pp. 363–373.   145

[75] J.C. Reynolds, *Types, abstraction, and parametric polymorphism*. In: **Information Processing 1983** (ed. R.E.A.Mason), North-Holland, Amsterdam, pp. 513–523.   145, 148

[76] David E. Rydeheard, *Functors and Natural Transformations*. In: **Category Theory and Computer Programming 1985**, LNCS 240, pp. 43–57.   50, 144

[77] D.S. Scott, *Domains for denotational semantics*. In: **Automata, Languages and Programming 1982** (ed. M.Nielsen, E.M.Schmidt), LNCS 140, pp. 577–613.   98

[78] M. Sintzoff, M. Weber, Ph. de Groote, J. Cazin, *Definition 1.1 of the generic development language Deva*. ToolUse-project, Research report, December 1991, Unité d'Informatique, Université Catholique de Louvain, Belgium.   12

[79] M.B. Smyth and G.D. Plotkin, *The Category-theoretic Solution of Recursive Domain Equations*. **Siam Journal of Computing 11** (1982), pp. 761–783.   98

[80] C. Spector, *Provably recursive functionals of analysis: a consistency proof of analysis by an extension of principles formulated in current intuitionistic mathematics*. In: **Proc. Symp. in Pure Mathematics V** (ed. J.C.E.Dekker), AMS, Providence 1962, pp. 1–27.   112

[81] A.S. Troelstra and D. van Dalen, *Constructivism in Mathematics*. Studies in Logic and the Foundation of Mathematics 123 and 125, North-Holland 1988.

[82] David Turner, *A New Formulation of Constructive Type Theory*. In: **Workshop on Programming Logic 1989**, report 54, Programming Methodology Group, Göteborg, pp. 258–294.

[83] Phil Wadler, *Theorems for free!*. In: **Functional Programming Languages and Computer Architecture 1989** (London), ACM Press, pp. 347–359.   144, 147, 148

[84] Matthias Weber, *Formalization of the Bird-Meertens Algorithmic Calculus in the Deva Meta-Calculus*. In: **Programming Concepts and Methods** (ed. Broy and Jones), North Holland 1990, pp. 201–232.   12

[85] M. Weber, M. Simons, C. Lafontaine, *The Generic Development Language DEVA*. LNCS 738 (1993).   12, 27, 30

[86] A. van Wijngaarden e.a., *Revised Report of the Algorithmic Language Algol 68*. Springer Verlag 1976.   19

[87] Martin Wirsing, *Algebraic Specification*. In: **Handbook of Theoretical Computer Science** (ed. J.van Leeuwen), Elsevier 1990, pp. 675–788.   63, 90

Note. 'LNCS' refers to the series "Lecture Notes in Computer Science", Springer-Verlag, Berlin.