The IMPS Bibliography

William M. Farmer*

Computing and Software
McMaster University
1280 Main Street West
Hamilton, ON L8S 4L7
Canada

Joshua D. Guttman†

Department of Computer Science
Worcester Polytechnic Institute
100 Institute Road
Worcester, MA 01609-2280
USA

F. Javier Thayer‡

The MITRE Corporation
202 Burlington Road
Bedford, MA 01730-1420
USA

14 November 2009

References

[1] W. M. Farmer. Abstract data types in many-sorted second-order logic. Technical Report M87-64, The MITRE Corporation, 1987.

[2] W. M. Farmer. A partial functions version of Church's simple theory of types. *Journal of Symbolic Logic*, 55:1269–91, 1990.

[3] W. M. Farmer. A simple type theory with partial functions and subtypes. *Annals of Pure and Applied Logic*, 64:211–240, 1993.

[4] W. M. Farmer. A general method for safely overwriting theories in mechanized mathematics systems. Technical report, The MITRE Corporation, 1994.

*wmfarmer@mcmaster.ca
†guttman@wpi.edu
‡jt@mitre.org

[5] W. M. Farmer. Theory interpretation in simple type theory. In J. Heering et al., editor, *Higher-Order Algebra, Logic, and Term Rewriting*, volume 816 of *Lecture Notes in Computer Science*, pages 96–123. Springer-Verlag, 1994.

[6] W. M. Farmer. Reasoning about partial functions with the aid of a computer. *Erkenntnis*, 43:279–294, 1995.

[7] W. M. Farmer. Mechanizing the traditional approach to partial functions. In W. Farmer, M. Kerber, and M. Kohlhase, editors, *CADE-13 Workshop on the Mechanization of Partial Functions*, pages 27–32, 1996.

[8] W. M. Farmer. The interactive mathematics laboratory. In *Proceedings of the 31st Annual Midwest Instruction and Computing Symposium (MICS '98)*, pages 16–18, April 1998.

[9] W. M. Farmer. STMM and partial functions. In M. Kerber, editor, *CADE-15 Workshop on the Mechanization of Partial Functions*, pages 3–14, 1998.

[10] W. M. Farmer. A scheme for defining partial higher-order functions by recursion. In *3rd Irish Workshop on Formal Methods*, electronic Workshops in Computing, `http://ewic.org.uk/workshops`, 1999.

[11] W. M. Farmer. An infrastructure for intertheory reasoning. In D. McAllester, editor, *Automated Deduction—CADE-17*, volume 1831 of *Lecture Notes in Computer Science*, pages 115–131. Springer-Verlag, 2000.

[12] W. M. Farmer. A proposal for the development of an interactive mathematics laboratory for mathematics education. In E. Melis, editor, *CADE-17 Workshop on Deduction Systems for Mathematics Education*, pages 20–25, 2000.

[13] W. M. Farmer. STMM: A set theory for mechanized mathematics. *Journal of Automated Reasoning*, 26:269–289, 2001.

[14] W. M. Farmer. Formalizing undefinedness arising in calculus. In D. Basin and M. Rusinowitch, editors, *Automated Reasoning—IJCAR 2004*, volume 3097 of *Lecture Notes in Computer Science*, pages 475–489. Springer-Verlag, 2004.

[15] W. M. Farmer. A sound and complete proof system for STTwU. Technical Report No. CAS-04-01-WF, McMaster University, 2004.

[16] W. M. Farmer. IMPS. In F. Wiedijk, editor, *The Seventeen Provers of the World*, volume 3600 of *Lecture Notes in Computer Science*, pages 72–87. Springer-Verlag, 2006.

[17] W. M. Farmer and O. Grigorov. Panoptes: An exploration tool for formal proofs. *Electronic Notes in Theoretical Computer Science*, 226:39–48, 2009. DOI:10.1016/j.entcs.2008.12.096.

[18] W. M. Farmer and J. D. Guttman. A set theory with support for partial functions. *Studia Logica*, 66:59–78, 2000.

[19] W. M. Farmer, J. D. Guttman, M. E. Nadel, and F. J. Thayer. Proof script pragmatics in IMPS. In A. Bundy, editor, *Automated Deduction—CADE-12*, volume 814 of *Lecture Notes in Computer Science*, pages 356–370. Springer-Verlag, 1994.

[20] W. M. Farmer, J. D. Guttman, and F. J. Thayer. IMPS: An Interactive Mathematical Proof System (system abstract). In M. E. Stickel, editor, *10th International Conference on Automated Deduction*, volume 449 of *Lecture Notes in Computer Science*, pages 653–654. Springer-Verlag, 1990.

[21] W. M. Farmer, J. D. Guttman, and F. J. Thayer. IMPS: System description. In D. Kapur, editor, *Automated Deduction—CADE-11*, volume 607 of *Lecture Notes in Computer Science*, pages 701–705. Springer-Verlag, 1992.

[22] W. M. Farmer, J. D. Guttman, and F. J. Thayer. Little theories. In D. Kapur, editor, *Automated Deduction—CADE-11*, volume 607 of *Lecture Notes in Computer Science*, pages 567–581. Springer-Verlag, 1992.

[23] W. M. Farmer, J. D. Guttman, and F. J. Thayer. IMPS: An Interactive Mathematical Proof System. *Journal of Automated Reasoning*, 11:213–248, 1993.

[24] W. M. Farmer, J. D. Guttman, and F. J. Thayer. The IMPS user's manual. Technical Report M-93B138, The MITRE Corporation, 1993.

[25] W. M. Farmer, J. D. Guttman, and F. J. Thayer. Reasoning with contexts. In A. Miola, editor, *Design and Implementation of Symbolic Computation Systems*, volume 722 of *Lecture Notes in Computer Science*, pages 216–228. Springer-Verlag, 1993.

[26] W. M. Farmer, J. D. Guttman, and F. J. Thayer. Contexts in mathematical reasoning and computation. *Journal of Symbolic Computation*, 19:201–216, 1995.

[27] W. M. Farmer, J. D. Guttman, and F. J. Thayer Fábrega. IMPS: An updated system description. In M. McRobbie and J. Slaney, editors, *Automated Deduction—CADE-13*, volume 1104 of *Lecture Notes in Computer Science*, pages 298–302. Springer-Verlag, 1996.

[28] W. M. Farmer and F. J. Thayer. Two computer-supported proofs in metric space topology. *Notices of the American Mathematical Society*, 38:1133–1138, 1991.

[29] W. M. Farmer and F. J. Thayer. Formal numerical program analysis. Technical report, The MITRE Corporation, 1994.

[30] W. M. Farmer and M. v. Mohrenschildt. Transformers for symbolic computation and formal deduction. In S. Colton, U. Martin, and V. Sorge, editors, *CADE-17 Workshop on the Role of Automated Deduction in Mathematics*, pages 36–45, 2000.

[31] O. Grigorov. Panoptes: An exploration tool for formal proofs. Master's thesis, McMaster University, 2008.

[32] J. D. Guttman. A proposed interface logic for verification environments. Technical Report M91-19, The MITRE Corporation, 1991.

[33] J. D. Guttman. Building verification environments from components: A position paper. In *Proceedings, Workshop on Effective Use of Automated Reasoning Technology in System Development*, pages 4–17, Naval Research Laboratory, Washington, D.C., April 1992.

[34] J. D. Guttman. A simple virtual memory scheme formalized in IMPS. Technical report, The MITRE Corporation, 1994.

[35] J. D. Guttman and D. M. Johnson. Three applications of Formal Methods at MITRE. In M. Naftalin, T. Denvir, and M. Bertran, editors, *FME '94: Industrial Benefits of Formal Methods*, volume 873 of *Lecture Notes in Computer Science*, pages 55–65. Springer Verlag, 1994.

[36] Y. Li. IMPS to OMDoc translation. Master's thesis, McMaster University, 2002.

[37] D. Miller. Two formal theories of character strings. Master's thesis, McMaster University, 2002.

[38] L. G. Monk. PDLM: A Proof Development Language for Mathematics. Technical Report M86-37, The MITRE Corporation, 1986.

[39] L. G. Monk. Inference rules using local contexts. *Journal of Automated Reasoning*, 4:445–462, 1988.

[40] J. A. Rees, N. I. Adams, and J. R. Meehan. *The T Manual*. Computer Science Department, Yale University, fifth edition, 1990.

[41] P. Tan. Mechanical verification of machine integer programs in a fragment of C. Master's thesis, McMaster University, 2002.

[42] F. J. Thayer. Obligated term replacements. Technical Report MTR-10301, The MITRE Corporation, 1987.

[43] F. J. Thayer. An approach to process algebra using IMPS. Technical Report MP-94B193, The MITRE Corporation, 1994.

[44] F. J. Thayer and J. D. Guttman. Copy on write. Technical report, The MITRE Corporation, 1995.