

CURRICULUM VITAE

Chad E. Brown Programming Systems Lab Universität des Saarlandes Campus E1 3 66123 Saarbrücken	 	cebrown@mathgate.info mathgate.info/cebrown/
--	--------------------------	---

Date and place of birth: Tupelo, Mississippi, June 14, 1970

Nationality: American

Education

Ph.D., Mathematics	August 2004	Carnegie Mellon University
M.S., Mathematics	May 1997	Purdue University
M.A., Mathematics	August 1994	Mississippi State University
B.S., Mathematics	December 1992	Mississippi State University

Career/Employment

Universität des Saarlandes	Wissensch. Mitarbeiter	August 2004 – present
Carnegie Mellon University	Research Assistant	September 1999 – July 2004
Universität des Saarlandes	Visiting Scientist	August 2001; August 2002
Carnegie Mellon University	Teaching Assistant	August 1997 – May 1999
Purdue University	Teaching Assistant	August 1995 – May 1997
Mississippi State University	Teaching Assistant	January 1993 – May 1994
Cypress Semiconductor	Programmer	January 1990 – March 1991

Specialization

- (i) **Main Field:** Higher-Order Theorem Proving
- (ii) **Subfields:** Semantics of Higher-Order Logic, Generalizations of Henkin Models, Extensionality Principles, Set Comprehension Principles, Cut-Elimination, Completeness and Independence Results, Automatic and Interactive Theorem Proving, Proof Checking, Proof Representations and Transformations, Categorical Semantics, Dependent Type Theory, Axiomatic Set Theory, Dependently Typed Set Theory, Mechanization of Mathematics, Mathematical Tutoring Systems
- (iii) **Current Interests:** Higher-Order Abstract Syntax, M-Sets, Tableaux, Functional Programming, Proof Carrying Code

Computing Skills

- **Extensive Experience:** Common Lisp, \LaTeX , TPS, Twelf, Scunak
- **Moderate Experience:** OCaml, Coq, Isabelle, Mizar, JavaScript, PHP, MySQL, HTML
- **Some Experience:** Standard ML, C, Haskell, Java, PERL

Publications

A. Books:

- (1) Chad E. Brown, *Automated Reasoning in Higher-Order Logic: Set Comprehension and Extensionality in Church's Type Theory*. *Studies in Logic: Logic and Cognitive Systems*, Volume 10. College Publications. (2007)

B. Ph.D. Thesis:

- (1) Chad E. Brown, *Set Comprehension in Church's Type Theory*, Ph.D. Thesis, Department of Mathematical Sciences, Carnegie Mellon University (2004).

C. Articles in journals:

- (1) V. Anuradha, Chad E. Brown, Ratnasingham Shivaji. Explosive non-negative solutions to two point boundary value problem, *J. Nonlinear Analysis, TMA* 26(3), 1996, pp. 613-630.
- (2) Peter B. Andrews, Chad E. Brown, F. Pfenning, Matthew Bishop, Sunil Issar, Hongwei Xi. ETPS: A System to Help Students Write Formal Proofs, *Journal of Automated Reasoning*, vol. 32, 2004, pp. 75-92.
- (3) Christoph E. Benzmüller, Chad E. Brown, Michael Kohlhase. Higher-Order Semantics and Extensionality, *Journal of Symbolic Logic*, vol. 69, 2004, pp. 1027-1088.
- (4) Peter B. Andrews, Chad E. Brown. TPS: A Hybrid Automatic-Interactive System for Developing Proofs, *Journal of Applied Logic*, 2006, 367-395.

D. Published contributions to academic conferences:

- (1) Peter B. Andrews, Chad E. Brown. System Description: TPS: A Theorem Proving System for Type Theory, 17th International Conference on Automated Deduction, Springer, LNAI 1831, 2000, pp. 164-169.
- (2) Peter B. Andrews, Chad E. Brown. Tutorial: Using TPS for Higher-Order Theorem Proving and ETPS for Teaching Logic, 17th International Conference on Automated Deduction, Springer, LNAI 1831, 2000, pp. 511-512.
- (3) Chad E. Brown. Solving for Set Variables in Higher-Order Theorem Proving, 18th International Conference on Automated Deduction, Springer, LNAI 2392, 2002, pp. 408-422.
- (4) Chad E. Brown. Reasoning in Extensional Type Theory with Equality. International Conference on Automated Deduction, Springer, LNAI 3632, 2005, pp. 23-37.
- (5) Christoph E. Benzmüller, Chad E. Brown. A Structured Set of Higher-Order Problems. Proceedings of the 18th International Conference on Theorem Proving in Higher Order Logics (TPHOLs 2005), LNAI vol. 3606, pp. 66-81, Oxford, UK, 2005. Springer.

- (6) Christoph E. Benzmüller, Chad E. Brown, Michael Kohlhase. Cut-Simulation in Impredicative Logics. Automated Reasoning, Third International Joint Conference, IJCAR 2006, Springer, LNAI 4130, 2006, pp. 220-234.
- (7) Chad E. Brown. Combining Type Theory and Untyped Set Theory. Automated Reasoning, Third International Joint Conference, IJCAR 2006, Springer, LNAI 4130, 2006, pp. 205-219.
- (8) Chad E. Brown. Verifying and Invalidating Textbook Proofs using Scunak. Mathematical Knowledge Management, MKM 2006, Springer, LNAI 4108, 2006, pp. 110-123.
- (9) Feryal Fulya Horozal, Chad E. Brown. Formal Representation of Mathematics in a Dependently Typed Set Theory. Mathematical Knowledge Management, MKM 2007, p. 265-279.

E. Published contributions to workshops:

- (1) Chad E. Brown. Encoding Functional Relations in Scunak. LFMTTP 2006. Seattle, Washington.

F. Articles in collections:

- (1) Christoph E. Benzmüller, Chad E. Brown. The Curious Inference of Boolos in Mizar and OMEGA. From Insight to Proof – Festschrift in Honour of Andrzej Trybulec. Studies in Logic, Grammar and Rhetoric 10(23), 2007, pp. 299-386.
- (2) Christoph E. Benzmüller, Chad E. Brown, Michael Kohlhase. Cut Elimination with xi-Functionality. Reasoning in Simple Type Theory: Festschrift in Honor of Peter B. Andrews on His 70th Birthday, 2008. *to appear*
- (3) Chad E. Brown. M-Set Models. Reasoning in Simple Type Theory: Festschrift in Honor of Peter B. Andrews on His 70th Birthday, 2008. *to appear*

G. Abstracts in journals:

- (1) Peter B. Andrews, Chad E. Brown. Proving theorems and teaching logic with TPS and ETPS, Bulletin of Symbolic Logic 11 (2005) 108–109, (abstract of contributed talk to the 2004 annual meeting of the Association for Symbolic Logic).
- (2) Chad E. Brown. Set comprehension in Church’s type theory, Bulletin of Symbolic Logic 11 (2005) 109, (abstract of contributed talk to the 2004 annual meeting of the Association for Symbolic Logic).

H. Others:

- (1) Chad E. Brown. Dependently Typed Set Theory. SEKI Working Paper, SWP-2006-03, 2006.
- (2) Chad E. Brown. Scunak Users Manual. 2006.

- (3) Christoph E. Benz Müller, Chad E. Brown, Michael Kohlhase. Higher-Order Semantics and Extensionality, Technical Report, CMU-03-001, 2003. ¹
- (4) Peter B. Andrews, Matthew Bishop, Chad E. Brown, Sunil Issar, Daniel Nesmith, Frank Pfenning, Hongwei Xi. TPS User's Manual. 2004. ²
- (5) Peter B. Andrews, Matthew Bishop, Chad E. Brown, Sunil Issar, Daniel Nesmith, Frank Pfenning, Hongwei Xi. ETPS User's Manual. 2004. ²
- (6) Peter B. Andrews, Matthew Bishop, Chad E. Brown, Sunil Issar, Daniel Nesmith, Frank Pfenning, Hongwei Xi. TPS Programmer's Guide, 2004.
- (7) Chad E. Brown. The Standard PER Model in the Reynolds/Ma Framework. 1999. ¹
- (8) Chad E. Brown. The Bainbridge-Freyd-Scedrov-Scott Parametric PER Model in the Reynolds/Ma Framework. 1999. ¹
- (9) Chad E. Brown. The Kolmogorov Translation of Classical Logic in Intuitionistic Logic. 1998. ¹

¹Available from <http://mathgate.info/cebrown/papers.php>

²Available from <http://gtps.math.cmu.edu/tps.html>

Teaching Experience

A. Courses

- (1) Lecture course in Summer 2008 at Saarland University: Introduction to Computational Logic. Colectured with Prof. Gert Smolka
- (2) Lecture course in Winter 2007 at Saarland University: Semantics of Classical Higher-Order Logic (SEMHOL)
- (3) Lecture course in Summer 2006 at ESSLLI 2006 in Malaga: Semantics of Higher-Order Logic (SEMHOL). Colectured with Dr. Christoph E. Benzmler
- (4) Lecture course in Winter 2006 at Saarland University: Semantics and Mechanization of Classical Higher-Order Logic (SEMHOL) Colectured with Dr. Christoph E. Benzmler
- (5) Lecture course in Summer 2005 at Saarland University: Introduction to Artificial Intelligence (AI) Colectured with Dr. Christoph E. Benzmler, Prof. J. Siekmann, Dr. Serge Autexier
- (6) Lecture course in Winter 2004 at Saarland University: Mathematical Assistance Systems (MAS) Colectured with Dr. Christoph E. Benzmler, Prof. J. Siekmann, Dr. Serge Autexier, Dr. C.-P. Wirth
- (7) Summer Course Instructor in Summer 1998 at Carnegie Mellon University for Integration, Differential Equations and Approximation (Student Evaluations Available)
- (8) Course in Spring 1994 at Mississippi State University: LISP Programming Lab
- (9) Course in Fall 1993 at Mississippi State University: LISP Programming Lab

B. Recitations and Tutorials

- (1) Peter B. Andrews, Chad E. Brown. Tutorial: Using TPS for Higher-Order Theorem Proving and ETPS for Teaching Logic. CADE 511-512, 2000. Colectured with Prof. Peter B. Andrews
- (2) Instructor for Two Recitations in Spring 1999 at Carnegie Mellon University for Calculus in Three Dimensions (Student Evaluations Available)
- (3) Instructor for Two Recitations in Fall 1998 at Carnegie Mellon University for Multivariate Analysis and Approximations (Student Evaluations Available)
- (4) Instructor for Two Recitations in Spring 1998 at Carnegie Mellon University for Integration, Differential Equations and Approximation (Student Evaluations Available)

- (5) Instructor for Two Recitations in Fall 1997 at Carnegie Mellon University for Integration and Differential Equations (Student Evaluations Available)
- (6) Instructor for Several Recitations at Purdue University from Fall 1995 to Spring 1997 for Calculus and Algebra Courses
- (7) Tutorial in Spring 1994 at Mississippi State University for a prealgebra course
- (8) Tutorial in Fall 1993 at Mississippi State University for a prealgebra course

C. Seminars

- (1) Summer 2005: Mathematical Assistant System Shootout

Coding Experience

Code written as parts of the following systems:

- TPS (a higher-order theorem prover in Common Lisp developed by the group of Peter B. Andrews)
- ETPS (educational version of TPS)
- TPS Java Interface
- Scunak 1.0 (a theorem prover for set theory in a dependent type theory written in Common Lisp)
- Jitpro (interactive higher-order theorem prover in JavaScript)
- Simptcheck (a proof checker in OCaml)
- JSIRSRPNLO (a small JavaScript program producing natural language proofs of the irrationality of square roots of numbers)

Summary:

- Total Lines of JavaScript: 4,702
- Total Lines of Common Lisp: 62,548
- Total Lines of OCaml: 1,337
- Total Lines of ocamllex: 36
- Total Lines of ocaml yacc: 46
- Total Lines of Java: 3,946