Curriculum Vitæ **Hongwei Xi** February 25, 2010

Contents

1 General Information

1.1 Personal Information

Associate Professor of Computer Science
Computer Science Department
111 Cummington Street
Boston University
Boston, MA 02215

Homepage: http://www.cs.bu.edu/~hwxi
Email: hwxi@cs.bu.edu
Phone: +1 617-358-2511
Fax: +1 617-353-6457

1.2 Short Biography

Hongwei Xi is currently an Associate Professor of Computer Science at Boston University. He received his Ph.D degree in Pure & Applied Logic from Carnegie Mellon University in 1998 and then worked from Sep. 1998 to Aug. 1999 as a post-doctoral researcher at Pacific Software Research Center, Department of Computer Science and Engineering, Oregon Graduate Institute. In 1999-2001, he was appointed an assistant professor in Department of Electrical and Computer Engineering and Computer Science, University of Cincinnati. He joined Computer Science Department at Boston University in October 2001. Dr. Xi's primary research focuses on the design and implementation of programming languages. In addition, he has developed keen interests to promoting software engineering benefits through the application of advanced type theory. At present, he has authored over 40 scientific papers and is the principal designer and implementer of the programming language ATS. He is a recipient of the National Science Foundation CAREER and ITR awards. He has also served as a program committee member for top programming language conferences such as ACM Symposium on Principles of Programming Languages (POPL), ACM Conference on Programing Language Design and Implementation (PLDI) and ACM International Conference on Functional Programming (ICFP).

1.3 Education

PhD in Pure & Applied Logic, December 1998, Carnegie Mellon University, Pittsburgh, USA
Thesis Title: *Dependent Types in Practical Programming*

M.S. in Mathematics, July 1988, Nanjing University, Nanjing, China
Thesis Title: *Half-Closed Intervals of Recursively Enumerable Degrees*

B.S. in Mathematics June 1985, Nanjing University, Nanjing, China

1.4 Academic Appointments

Sep. 2007 to Present	Associate Professor, Computer Science Department, Boston University
Oct. 2001 to Aug. 2007	Assistant Professor, Computer Science Department, Boston University
Sep. 1999 to Oct. 2001	Assistant Professor, Department of Electrical and Computer Engineering and Computer Science, University of Cincinnati
Aug. 1998 to Aug. 1999	Postdoctoral Research Associate, Pacific Software Research Center, Department of Computer Science and Engineering Oregon Graduate Institute
Jul. 1988 to Jun. 1992	Lecturer, Department of Computer Science, Shanghai Jiao Tong University

1.5 Honors

- National Science Foundation CAREER Award, 2001.

1.6 Professional and Scholarly Associations

Association of Computing Machinery (ACM), Institute of Electrical and Electronics Engineers (IEEE), American Mathematical Society (AMS)

2 Research

In August 2006, Hongwei Xi was ranked the 6212^{th} most cited author in computer science (among 790,329 authors) by the NECI Scientific Digital Library, available at

`http://citeseer.ist.psu.edu/allcited.html`

2.1 Research Summary

Hongwei Xi's primary research focus is on the design and implementation of programming languages. In addition, he has developed keen interests to promoting software engineering benefits through the application of advanced type theory. Ideally, a programming language should be simple and general, and it should permit extensive error checking, facilitate proofs of program properties and possess a correct and efficient implementation. There will invariably be conflicts among these goals that must be resolved with careful attention to the needs of the user. In order to make significant progress, it is necessary to adopt approaches that are capable of addressing *realistic* problems effectively.

Dependent Type Systems Hongwei Xi's doctoral thesis studies an enrichment of ML with a form of dependent types, where type index expressions are restricted to a pure type index language $\mathcal{L}$, leading to the $DML(\mathcal{L})$ language schema. The aim of this study was mainly to provide for specification and inference of significantly more precise type information compared with the current type system of ML, facilitating program error detection and compiler optimization. It has since been shown that type-checking a sufficiently annotated program in $DML(\mathcal{L})$ can be reduced to constraint satisfaction in the type index language $\mathcal{L}$. Therefore, type-checking in $DML(\mathcal{L})$ can be made practical for those type index languages $\mathcal{L}$ for which efficient constraint solvers can be provided. Various examples have been verified in a prototype implementation of a type-checker for $DML(\mathcal{L})$, where $\mathcal{L}$ is a type index language in which constraints are linear inequalities on integers. With Robert Harper, Xi also studied the use of dependent types at assembly level, forming a dependently typed assembly language. In addition, he has incorporated dependent types into imperative programming by forming a dependently typed imperative language Xanadu.

Guarded Recursive Datatypes Hongwei Xi introduced a notion of guarded recursive datatypes into programming language design, demonstrating that many programming features (e.g., object-oriented programming, modular programming, meta-programming) can be supported naturally through the use of guarded recursive datatypes in a typeful manner. This notion, which is now given the name *generalized algebraic datatype*, has since been widely adopted in the functional programming language community and beyond.

Applied Type System Currently, Hongwei Xi is in the process of designing and implementing a programming language ATS, which is equipped with a highly expressive type system rooted in

the framework Applied Type System (ATS). In ATS, a variety of programming paradigms are supported in a typeful manner, which include: functional programming (available), object-oriented programming (available), imperative programming with pointers (available), modular programming (available), meta programming (available), assembly programming (under development). Also, ATS contains a component ATS/LF that supports a form of (interactive) theorem proving. With this component, a programming style that combines programming with theorem proving can be readily supported. Furthermore, this component can also be used as a logical framework to encode various deduction systems and their properties.

2.2 Publications

Refereed Journal Articles

[J1] Hongwei Xi, Dependent ML: an approach to practical programming with dependent types, Journal of Functional Programming, vol. 17(2), pp. 215-286, 2007.

[J2] Chiyan Chen, Rui Shi and Hongwei Xi, Implementing Typeful Program Transformations, Fundamenta Informaticae, vol. 69(1-2), pp. 103-121, 2005.

[J3] Chiyan Chen and Hongwei Xi, Meta-Programming through Typeful Code Representation, Journal of Functional Programming, vol. 15(6), pp. 797–835, 2005.

[J4] Peter B. Andrews, Matthew Bishop, Chad Brown, Sunil Issar, Frank Pfenning and Hongwei Xi, ETPS: A System to Help Students Write Formal Proofs, Journal of Automated Reasoning (JAR), vol. 32(1), pp. 75–92, 2004.

[J5] Hongwei Xi, Dependently Typed Pattern Matching, Journal of Universal Computer Science (JUCS), vol. 9(8), pp. 851-872, August 2003.

[J6] Hongwei Xi, Dependent Types for Program Termination Verification, Journal of Higher-Order Symbolic Computation, vol. 15(1), pp. 91–131, March 2002.

[J7] Femke van Raamsdonk, Paula Severi, Morten H. Sorensen and Hongwei Xi, Perpetual Reductions in Lambda-Calculus, Journal of Information and Computation, vol. 149(2), pp. 173–225, March 1999.

[J8] Hongwei Xi, Upper bounds for standardization and an application, Journal of Symbolic Logic, vol. 64(1), pp. 291–303, March 1999.

[J9] Peter B. Andrews, Matthew Bishop, Sunil Issar, Dan Nesmith, Frank Pfenning, and Hongwei Xi, TPS: A Theorem Proving System for Classical Type Theory, Journal of Automated Reasoning, vol. 16(3), pp. 321–353, 1996.

[J10] Hongwei Xi, Half-Closed Intervals of Recursively Enumerable Degrees (in Chinese with English abstract), Math. Semiannuals of Nanjing University, P.R. China, 1989.

Refereed Conference Papers

[C1] Rui Shi, Chiyan Chen and Hongwei Xi, Distributed Meta-Programming. In Proceedings of the 5th International Conference on Generative Programming and Component Engineering (GPCE'06), Portland, OR, October 2006.

[C2] Chiyan Chen and Hongwei Xi, Combining Programming with Theorem Proving. In Proceedings of the 10th International Conference on Functional Programming (ICFP'05), Tallinn, Estonia, September 2005.

[C3] Dengping Zhu and Hongwei Xi, Safe Programming with Pointers through Stateful Views. In Proceedings of the 7th International Symposium on Practical Aspects of Declarative Languages (PADL'05), Springer-Verlag LNCS vol. 3350, pp. 83–97, Long Beach, CA, January 2005.

[C4] Chiyan Chen, Rui Shi and Hongwei Xi, A Typeful Approach to Object-Oriented Programming with Multiple Inheritance. In Proceedings of the 6th International Symposium on Practical Aspects of Declarative Languages (PADL'04), Springer-Verlag LNCS vol. 3057, pp. 23–38, Dallas, TX, June 2004.

[C5] Chiyan Chen, Dengping Zhu and Hongwei Xi, Implementing Cut Elimination: A Case Study of Simulating Dependent Types in Haskell. In Proceedings of the 6th International Symposium on Practical Aspects of Declarative Languages (PADL'04), Springer-Verlag LNCS vol. 3057, pp. 239–254, Dallas, TX, June 2004.

[C6] Dengping Zhu and Hongwei Xi, A Typeful and Tagless Representation for XML Documents. In Proceedings of the First Asian Symposium on Programming Languages and Systems (APLAS'03), Springer-Verlag LNCS vol. 2895, pp. 89–104, Beijing, China, November 2003.

[C7] Walid Taha, Stephan Ellner and Hongwei Xi, Generating Imperative, Heap-Bounded Programs in a Functional Setting. In Proceedings of the Third International Conference on Embedded Software (EMSOFT'03), Springer-Verlag LNCS vol. 2855, pp. 340–355, Philadelphia, PA, October 2003.

[C8] Hongwei Xi, Facilitating Program Verification with Dependent Types. In Proceedings of International Conference on Software Engineering and Formal Methods (SEFM'03), pp. 72–81, Brisbane, Australia, September 2003.

[C9] Chiyan Chen and Hongwei Xi, Meta-Programming through Typeful Code Representation. In Proceedings of the 8th International Conference on Functional Programming (ICFP'03), pp. 275–286, Uppsala, Sweden, August 2003.

[C10] Hongwei Xi, Dependently Typed Pattern Matching. In Proceedings of Simposio Brasileiro de Linguagens de Programacao (SBLP'03), pp. 149–165, Ouro Preto, Brazil, May 2003.

[C11] Hongwei Xi, Chiyan Chen and Gang Chen, Guarded Recursive Datatype Constructors. In Proceedings of the 30th ACM SIGPLAN Symposium on Principles of Programming Languages (POPL'03), pp. 224–235, New Orleans, Louisiana, January 2003.

[C12] Hongwei Xi and Robert Harper, Dependently Typed Assembly Language. In Proceedings of the 6th International Conference on Functional Programming (ICFP'01), pp. 169–180, Florence, September 2001.

[C13] Hongwei Xi, Dependent Types for Program Termination Verification. In Proceedings of 16th Symposium on Logic in Computer Science (LICS'01), pp. 231–242, Boston, June 2001.

[C14] Hongwei Xi, Imperative Programming with Dependent Types, In Proceedings of 15th Symposium on Logic in Computer Science (LICS'00), pp. 375–387, Santa Barbara, June 2000.

[C15] Hongwei Xi and Songtao Xia, Towards Array Bound Check Elimination in Java Virtual Machine Language. In Proceedings of CASCON'99, pp. 110–125, Mississauga, Ontario, November 1999.

[C16] Hongwei Xi and Frank Pfenning, Dependent Types in Practical Programming. In Proceedings of ACM SIGPLAN Symposium on Principles of Programming Languages (POPL'99), pp. 214–227, San Antonio, January 1999.

[C17] Hongwei Xi and Frank Pfenning, Eliminating Array Bound Checking Through Dependent Types. In Proceedings of ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'98), pp. 249–257, Montreal, June 1998.

[C18] Hongwei Xi, Towards automated termination proofs through "Freezing". In the proceedings of 9th International Conference on Rewriting Techniques and Applications (RTA'98), pp. 271–285, Japan, April 1998.

[C19] Hongwei Xi, Evaluation under lambda-abstraction. In Proceedings of 9th International Symposium on Programming Languages, Implementations, Logics, and Programs (PLILP'97), Springer-Verlag LNCS 1292, pp. 259–273, Southampton, UK, September 1997.

[C20] Hongwei Xi, Upper bounds for standardization and an application. In the proceedings of Kurt Gdel Colloquium 1997 (KGC'97), Springer-Verlag LNCS, vol. 1289, pp. 335–348, September 1997.

[C21] Hongwei Xi, Simulating Eta-Expansions with Beta-Reductions in the Second-Order Polymorphic Lambda-Calculus. In the proceedings of Symposium on Logical Foundations of Computer Science (LFCS'97), Springer-Verlag LNCS, vol. 1234, pp. 399–409, Yaroslavl, July 1997.

[C22] Hongwei Xi, Weak and Strong Beta Normalizations in Typed Lambda-Calculi. In the proceedings of Typed Lambda Calculi and Applications (TLCA'97), Springer-Verlag LNCS, vol. 1210, pp. 390–404, April 1997.

Refereed Workshop Papers

[W1] Kevin Donnelly and Hongwei Xi, A Formalization of Strong Normalization for Simply Typed Lamdba-Calculus and System F, Workshop on Logic Frameworks and Meta Languages: Theory and Practice (LFMTP'06), Seattle, WA, August 2006.

[W2] Kevin Donnelly and Hongwei Xi, System Description: Combining Higher-Order Abstract Syntax with First-Order Abstract Syntax in ATS, Workshop on Mechanized Reasoning about Languages with Variable Binding (MERLIN'05), Tallinn, Estonia, September 2005.

[W3] Sa Cui and Kevin Donnelly and Hongwei Xi, ATS: a language that combines programming with theorem proving. In Proceedings of the 5th International Workshop on Frontiers of Combining Systems (FroCos'05), Springer-Verlag LNCS vol. 3717, pp. 310–320, Vienna, Austria, September 2005.

[W4] Hongwei Xi, Development Separation in Lambda-Calculus, In Proceedings of the 12th Workshop on Logic, Language, Information and Computation (WoLLIC'05), ENTCS vol. 143, pp. 207–221, Florianopolis, Santa Catarina, Brazil, July 2005.

[W5] Hongwei Xi, Applied Type System (extended abstract). In the post-workshop proceedings of TYPES 2003, Springer-Verlag LNCS vol. 3085, pp. 394–408, 2004.

[W6] Chiyan Chen and Hongwei Xi, Implementing Typeful Program Transformations, ACM SIGPLAN 2003 Workshop on Partial Evaluation and Semantics Based Program Manipulation (PEPM'03), pp. 20–28, San Diego, CA, June 2003.

[W7] Hongwei Xi and Carsten Schuermann, CPS Transform for Dependent ML (abstract). In the meeting report of the 8th Workshop on Logic, Language, Information and Computation (WoLLIC'01), Logic Journal of IGPL, 9(5), pp. 739–754, Brasilia, Brazil, August 2001.

[W8] Hongwei Xi, Dependently Typed Data Structures. In Proceedings of the Workshop on Algorithmic Aspects of Advanced Programming Languages (WAAAPL'99), pp. 17–32, Paris, September 1999.

[W9] Hongwei Xi, Dead Code Elimination through Dependent Types. In Proceedings of the First International Workshop on Practical Aspects of Declarative Languages(PADL'99), Springer-Verlag LNCS vol. 1551, pp.228–242, San Antonio, January 1999.

[W10] Hongwei Xi, Generalized Lambda-Calculi (abstract). In the meeting report of the 4th Workshop on Logic, Language, Information and Computation (WoLLIC'97), Logic Journal of IGPL, 5(6), pp. 925–927, Fortaleza (Ceara), Brazil, August 1997.

[W11] Peter B. Andrews, Matthew Bishop, Sunil Issar, Dan Nesmith, Frank Pfenning, and Hongwei Xi, TPS: An interactive and automatic tool for proving theorems of type theory, In Jeffrey J. Joyce and Carl-Johan H. Seger, editors, Proceedings of the 6th International Workshop on Higher Order Logic Theorem Proving and Its Applications, pages 366-370, Vancouver, B.C., Canada, August 1993. Springer-Verlag LNCS 780.

[W12] Hongwei Xi, On branching and nonbranching recursively enumerable degrees (in Chinese), National Logic Workshop, Shantou University, China, October, 1990.

Invited Conference Paper

[I1] Hongwei Xi, Unifying Object-Oriented Programming with Typed Functional Programming, ASIAN Symposium on Partial Evaluation and Semantics-Based Program Manipulation (ASIA-PEPM'02), pp. 117–125, Aizu-Wakamatsu, Japan, September 2002.

Technical Reports

1. Hongwei Xi and Dengping Zhu and Yanka Li, Applied Type System with Stateful Views. Technical Report, no. BUCS-2005-03, Computer Science Department, Boston University, January, 2005.

2. Hongwei Xi and Joachim Steinbach, Erasure for Termination Proofs, Technical Report OGI-CSE-99-009, Department of Computer Science and Engineering, Oregon Graduate Institute, August 1999. (My research interests have since left the area of rewriting systems)

3. Joachim Steinbach and Hongwei Xi. Freezing – Termination for Classical, Context-Sensitive and Innermost Rewriting, Institut für Informatik, Technische Universitat Muenchen, January, 1998. (My research interests have since left the area of rewriting systems)

4. Hongwei Xi. An Induction Measure on Lambda-Terms and Its Applications. Research Report 96-192, Department of Mathematical Sciences, 1996.

5. Hongwei Xi. On Weak and Strong Normalizations. Research Report 96-187, Mathematics Department, Carnegie Mellon University, 1996.

Technical Writing

1. Hongwei Xi, Programming in ATS. (about 70 pages now) (This writing serves as a Programmer's Manual for programming in ATS, a language that I am currently designing and implementing).

2. Dan Nesmith, Matthew Bishop, Peter Andrews, Sunil Issar, Frank Pfenning, and Hongwei Xi, TPS User's Manual, pp. 84+iii.

3. Peter Andrews, Sunil Issar, Dan Nesmith, Frank Pfenning, Hongwei Xi and Matthew Bishop, TPS3 Facilities Guide for Programmers and Users, pp. 238+viii.

4. Frank Pfenning, Sunil Issar, Dan Nesmith, Peter Andrews, Hongwei Xi and Matthew Bishop, ETPS User's Manual, Version for Mathematical Logic I & II, pp. 60+ii.

2.3 Talks

Invited Talks at International Meetings

- To memory safety through proofs and beyond, the International Conference of the TYPES Project (TYPES 2006), University of Nottingham, United Kingdom, 18-21 April 2006.

- Implementing an evaluator for mini-ML in ATS: a case of programming with theorem proving, Fifth International Workshop on Reduction Strategies in Rewriting and Programming (WRS'05), Nara, Japan, April 22, 2005.

- Unifying Object-Oriented Programming with Typed Functional Programming, ASIAN Symposium on Partial Evaluation and Semantics-Based Program Manipulation (ASIA-PEPM), Aizu-Wakamatsu, Japan, 14 September 2002.

- Compiling with Dependent Types, Workshop on Proof-Carrying Code, Santa Barbara, CA, 29 June 2000.

- Dependently Typed Assembly Language, Workshop on Dependent Types in Programming, Göteborg, Sweden, 27 - 28 March 1999.

Invited Departmental Colloquium Talks

- Programming with C library functions safely, Computer Science and Engineering Department, Washington University at St. Louis, MO, 11 November 2005.

- Implementing Staged Computation, Information Processing Laboratory, Department of Information Engineering and Department of Mathematical Science and Informatics, University of Tokyo, Tokyo, Japan, 11 September 2002.

Talks at Universities and Research Institutions

- To memory safety through proofs, Triforce group, DEAS, Harvard University, November 30, 2006.
- To memory safety through proofs, Sun Microsystems, Burlington, MA, 17 November 2005.
- ATS: a language to make typeful programming real and fun, Computer Science Department, Yale University, New Haven, CT, 28 April 2005.
- Unifying Object-Oriented Programming with Typed Functional Programming, Computer Science Department, Rice University, Houston, TX, 16 & 18 April 2003.
- Dependent Types for Program Termination Verification, Computer Science Department, Yale University, 13 June 2001.
- A Dependently Typed Assembly Language, ROPAS group, Department of Computer Science, Korea Advanced Institute of Science and Technology (KAIST), Taejon, Korea, 7 April 1999.
- Dependent Types in Practical Programming, ROPAS group, Department of Computer Science, Korea Advanced Institute of Science and Technology (KAIST), Taejon, Korea, 6 April 1999.
- Dependent Types in Practical Programming, Formal Methods PI meeting, Stanford University, October 1998.

More Talks The talks listed next were given at institutions with which Hongwei Xi was not affiliated at the time.

University of Toronto, Toronto (May, 2001); Princeton University, Princeton, NJ (Apr 2001); Northwestern University, Evanston, IL (Apr 2001); Boston University, Boston. MA (Mar 2001); University of Florida, Gainesville, FL (Mar 2001); University of Maryland, College Park, MD (Mar 2001); Syracuse University, Syracuse, NY (Mar 2001); University of Minnesota, Minneapolis, MN (Feb 2001); Worcester Polytechnic Institute, Worcester, MA (Feb 2001); Polytechnic University, New York, NY (Apr 1999); New Jersey Institute of Technology; Newark, NJ (Mar 1999); Iowa State University, Ames, IA (Mar 1999); University of New Hampshire, Durham, NH (Mar 1999); University of Cincinnati, Cincinnati, OH (Feb 1999); University of Nevada, Las Vegas (Feb 1999); Oklahoma State University, Stillwater, OK (Feb 1999); Ohio University, Athens, OH (August, 1998); Oregon Graduate Institute of Science and Technology (Jun 1998); York University, Toronto (Apr 1998); Singapore National University, Singapore (Mar 1998); Portland State University, Portland, OR (Mar 1998); etc.

Presentation at Conferences

- Combining Programming with Theorem Proving, the 10th International Conference on Functional Programming (ICFP'05), Tallinn, Estonia, September 2005.

- Meta-Programming through Typeful Code Representation, the 8th International Conference on Functional Programming (ICFP'03), Uppsala, Sweden, August 2003.
- Guarded Recursive Datatype Constructors, the 30th ACM SIGPLAN Symposium on Principles of Programming Languages (POPL'03), New Orleans, Louisiana, January 2003.
- A Dependently Typed Assembly Language, the 6th International Conference on Functional Programming (ICFP'01), pp. 169–180, Florence, September 2001
- Dependent Types for Program Termination Verification, the 16th IEEE SIGPLAN Symposium on Logic in Computer Science (LICS'01), Boston, MA, June 2001.
- Imperative Programming with Dependent Types, the 15th IEEE SIGPLAN Symposium on Logic in Computer Science (LICS'00), Santa Barbara, CA, June 2000.
- Dependent Types in Practical Programming, the 26th ACM SIGPLAN Symposium on Principles of Programming Languages (POPL'99), San Antonio, January 1999.
- Eliminating Array Bound Checking through Dependent Types, ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'98), Montreal, June, 1998.

2.4 Funding

Research Grants

- Hongwei Xi (PI), ATS: A Lanugage to Support Practical Programming with Theorem Proving, National Science Foundation, CCR-0702665, $299,980, 2007 - 2010.
- Hongwei Xi (Senior Personnel) with A. Bestavros, M. Betke, S. Sclaroff, M. Crovella, G. Kollios, I. Matta, G. Itkis, A. Kfoury, L. Reyzin, R. West, SENSORIUM: Research Infrastructure for Managing Spatio-Temporal Objects in Video Sensor Networks, National Science Foundation, EIA-0202067, $1,247,395, August 1, 2002 - July 31, 2007.
- Hongwei Xi (PI), CAREER: Realistic Program Termination Verification: Theory and Practice, National Science Foundation, CCR-0092703/0229480, $284,938, July 1, 2001 - June 30, 2006.
- Hongwei Xi (PI), Imperative Programming with Dependent Types, National Science Foundation, CCR-0081316/0229480, $334,918, CCR-0081316/0224244, September 1, 2000 - August 31, 2004.
- Paul Sivilotti (co-PI) and Hongwei Xi (co-PI), Dependent Types for High-Confidence Distributed Systems, Ohio Board of Regents, $89,954, 2000-2002. There is no PI for this grant.
- Hongwei Xi (PI), Faculty Development, University of Cincinnati, $3,108, Summer of 2000.

3 Teaching

3.1 Teaching at Boston University

Course Number	Course Name	Academic Year	Contact Hours	Number of Students	Additional Information
CS320	Concepts of Prog.	01/02	4.0	65	1 TF and 2 graders
CS520	Principles of Prog.	02/03	4.0	26	1 grader
CS591	Comp. & Ded.	02/03	4.0	08	
CS520	Principles of Prog.	03/04	4.0	11	1 grader
CS525	Compiler Design	03/04	4.0	14	1 grader
CS320	Concepts of Prog.	04/05	4.0	35	1 TF and 2 graders
CS525	Compiler Design	04/05	4.0	08	1 grader
CS792	Seminar	04/05	4.0	11	with Prof. Kfoury
CS520	Principles of Prog.	05/06	4.0	06	
CS112	Intro to CS II	05/06	4.0	13	1 TF and 1 grader
CS320	Concepts of Prog.	06/07	4.0	27	1 TF and 1 grader
CS112	Intro to CS II	06/07	4.0	07	1 TF and 1 grader
CS520	Principles of Prog.	08/09	4.0	15	1 grader
CS525	Compiler Design	08/09	4.0	9	
CS520	Principles of Prog.	09/10	4.0	7	
CS112	Intro to CS II	09/10	4.0	30	1 TF and 2 graders

3.2 Teaching at University of Cincinnati

Course Number	Course Name	Academic Year	Contact Hours	Number of Students
ECES603	Principles of Prog. Lang.	99/00	4.0	37
ECES717	Type Systems	99/00	4.0	6
ECES670	Automata and Form. Lang.	00/01	4.0	13
ECES403	Organization of Prog. Lang.	00/01	4.0	8

3.3 Teaching at Other Institutions

Hongwei Xi worked as an instructor for several calculus courses at Carnegie Mellon University during the summers of 1996, 1997 and 1998. He also taught Mathematical Logic and Discrete Mathematics when he was a lecturer at Shanghai Jiao Tong University.

3.4 Current Students

- Matthew Danish, PhD Student, Boston University, September 2008 - Present.

- Likai Liu, PhD Student, Boston University, September 2004(?) - Present. (Likai passed his PhD depth examination in Spring 2008).

3.5 Former Students

- Ph.D. Students
  - Rui Shi, PhD student, Boston University, September 2002 - May 2007. (Rui successfully defended his doctoral thesis titled *Types for Safe Resource Sharing in Sequential and Concurrent Programming* on May 3, 2007)
  - Dengping Zhu, PhD student, Boston University, September 2001 - May 2006. (Dengping successfully defended his doctoral thesis titled *To memory safety through proofs* on April 3, 2006)
  - Chiyan Chen, PhD student, Boston University, June 2002 - August 2005. (Chiyan successfully defended his doctoral thesis titled *Type Inference in Applied Type System* on July 29, 2005)
- Master Students
  - Michel Machado, Master student, Boston University, September 2006 - May 2008. (Michel successfully completed his thesis titled *A Dependent Type for Strings to Avoid Injections*)
  - Kevin Donnelly, Master student, Boston University, September 2004 - December 2007. (Kevin successfully completed his thesis titled *Strong Normalization for System FC*)
  - Rick Lavoie, Master student, Boston University, September 2005 - May 2007. (Rick successfully completed his project on building a bytecode interpreter and a native compiler for ATS).
  - Sa Cui, Master student, Boston University, September 2002 - May 2006. (Sa successfully defended his Master's thesis titled *Typeful Assembly Programming in ATS* on April 4, 2006)
  - Ye Jin, Master student, Boston University, January 2004 - May 2005. (Ye finished a Master's project on *Implementing Basic Category Theory in ATS* and graduated in May 2005)
  - Sudeep Sabnis, Master student, University of Cincinnati, January 2001 - August 2003. (Sudeep successfully defended his Master's thesis titled *An Approach to Facilitating Verification of Linear Constraints* in August 2003)
  - Varun Nayak, Master student, University of Cincinnati, January 2000 - May 2002. (Varun successfully defended his Master's thesis titled *A Survey on Algorithms for Solving Linear Integer Type Constraints* in May 2002 and is now working for a company in London, England)

4 Service

4.1 Service to Boston University

College of Arts & Sciences

- Natural science curriculum committee (Sep 2005 - May 2007)
- Faculty advisor during academic orientation (2007, 2006, 2005, 2004, 2003, 2002)

Computer Science Department

- Personnel supervisor of the system staff (Sep 2005 - July 2007)
- Member of the programming language in-depth examination committee (Dec 2005)
- Faculty annual report evaluation committee (Oct 2004)
- Departmental space coordinator (Sep 2004 - May 2005)
- Departmental colloquium coordinator (Sep 2003 - May 2004)
- Member (representing PL) of departmental graduate admissions committee, (Jan - May 2005)
- Member (representing PL) of departmental graduate admissions committee, (Jan - May 2004)

4.2 Conference and Workshop Program Committees

Co-Chair/Co-Organizer

- The 2nd workshop on Programming Languages meet Program Verification (PLPV'07), Freiberg, Germany, October, 2007 (co-chairing with Aaron Stump)
- The 1st workshop on Programming Languages meet Program Verification (PLPV'06), Seattle, Washington, August, 2006 (co-chairing with Aaron Stump)
- The 14th New England Programming Language and Systems (NEPLS), February, 2005 (co-organizing with Assaf Kfoury)

Program Committee Member

- The 2008 ACM SIGPLAN Workshop on ML, Victoria, British Columbia, Canada, September 21st, 2008
- The 3rd workshop on Logic and Semantic Frameworks, with Applications (LSFA'08), San Salvaldor, Brazil, September 2nd, 2007.

- ACM SIGPLAN Conference on Programming Language Design and Implementation (PLDI'08), Tucson, AZ, June, 2008
- The 2nd workshop on Logic and Semantic Frameworks, with Applications (LSFA'07), Ouro Preto, Brazil, August 28th, 2007.
- The 2nd International Conference on Software and Data Technologies (ICSOFT'07), Barcelona, Spain, July 22-25, 2007.
- ACM Workshop on Types in Language Design and Implementation (TLDI'07), Nice, France, January 16, 2007.
- The 11th ACM SIGPLAN International Conference on Functional Programming (ICFP'06), Portland, Oregon, September 18-20, 2006
- The 1st International Conference on Software and Data Technologies, (ICSOFT'06) Sétubal, Portugal, September 11-14, 2006.
- The 33rd ACM SIGPLAN-SIGACT Symposium on Principles of Progamming Languages (POPL'06), Charleston, South Carolina, January 11-13, 2006
- The Second International Conference on Embedded Software and Systems (ICESS'05), Xi'an, China, Dec 16-18, 2005.
- The 2nd MetaOCaml Workshop, Tallin, Estonia, September 28, 2005
- The 15th New England Programming Language and Systems (NEPLS), June, 2005
- The 2nd ASIAN Symposium on Progamming Languages and Systems (APLAS 2004), Taipei, Taiwan, November 4-6, 2004
- The 1st MetaOCaml Workshop, Vancouver, BC, Canada, October 25, 2004

External Reviewer

- Netherlands Organization for Scientific Research (NWO), 2005
- National Science Foundation, 2004.

4.3 Journal/Conference Submission Refereeing

International Conference on Compiler Construction (CC) (2008); European Symposium on Programming (ESOP) (2008, 2007, 2006, 2003); Foundations of Software Science and Computation Structures (FoSSaCS) (2007); Symposium on Principles of Programming Languages (POPL) (2008, 2007, 2006, 2005, 2004, 2003); Real-Time Systems Symposium (RTSS) (2005); Computer Science

Logic (CSL) (2005); Journal of Science of Computer Programming (SCP) (2005); International Colloquium on Automata, Languages and Programming (ICALP) (2005); IEEE Symposium on Logic in Computer Science (LICS) (2007, 2005, 2003, 2002, 2000); Journal of Functional Programming (JFP) (2007, 2005, 2004, 2001); Transaction on Programming Languages and Systems (TOPLAS) (2005, 2004, 2003, 2001); Typed Lambda-Calculus and Applications (TLCA) (2007, 2005); International Conference on Rewriting Techniques and Applications (RTA) (2007, 2006); Practical Aspects of Declarative Languages (PADL) (2005); Journal of Higher-Order Symbolic Computation (HOSC) (2004, 2001, 2000, 1999); ACM International Conference on Functional Programming (ICFP) (2006, 2004, 2003, 2002, 1998); Journal of Automated Reasoning (2004, 2003, 2002); Real-Time and Embedded Technology and Applications Symposium (RTAS) (2004); TYPES'03 Workshop (2003); Asian Symposium on Programming Languages and Systems (2004, 2003); The Bulletin of Symbolic Logic (2003); Haskell Workshop (2003); International Computing and Combinatorics Conference (COCOON) (2003); Workshop on Programs as Data Objects (PADO) (2000); The First International Conference on Parallel and Distributed Computing, Applications and Technologies (PDCAT) (2000); CSER (Consortium for Software Engineering Research) Book (2000); Journal of Information and Computation (1996); etc.