

Chau-Wen Tseng

1 PERSONAL INFORMATION

- Associate Professor, University of Maryland
Department of Computer Science
Appointed August, 1995.

EDUCATION

- Ph.D., Computer Science, Rice University, 1993.
Dissertation: *An Optimizing Fortran D Compiler for MIMD Distributed-Memory Machines*
Advisor: *Ken Kennedy*
- M.S., Computer Science, Rice University, 1992.
- A.B., Computer Science, Harvard University, 1986.

EMPLOYMENT

- Associate Professor, Computer Science Department, UMCP.
July 2001–*present*.
- Assistant Professor, Computer Science Department, UMCP.
August 1995–July 2001.
- Consultant, Silicon Graphics Computer Systems, Mountain View, California, 1995.
- Research Associate, Stanford University, Stanford, California, 1993–1995.
- Research Scientist, Rice University, Houston, Texas, 1993.
- Research Assistant, Rice University, Houston, Texas, 1988–1992.
- Software Design Engineer, Teradyne, Boston, Massachusetts, 1986–1988.
- Applications Programmer, Intentional Educations, Watertown, Massachusetts, 1984–1986.

2 RESEARCH, SCHOLARLY, AND CREATIVE ACTIVITIES²

a BOOKS

1. “Languages and Compilers for Parallel Computing, 15th Workshop, LCPC 2002, College Park, MD USA, July 2002, Revised Papers,” B. Pugh and C.-W. Tseng (Eds.), Lecture Notes in Computer Science, Volume 2481, Springer-Verlag, 2005.
2. “Advances in Computers: Computational Biology and Bioinformatics,” C.-W. Tseng (Ed.), Volume 68, Elsevier, 2006.

iii CHAPTERS IN BOOKS

1. “Compiler Support for Machine-Independent Parallel Programming in Fortran D,” S. Hiranandani, K. Kennedy, and C.-W. Tseng, *Compilers, and Run-Time Environments for Distributed Memory Machines* (J. Saltz and P. Mehrotra, eds.), North-Holland, Amsterdam, The Netherlands, 1992, pages 139–176.
2. “An Overview of the Fortran D Programming System,” S. Hiranandani, K. Kennedy, C. Koelbel, U. Kremer, and C.-W. Tseng, *Languages and Compilers for Parallel Computing, Fourth International Workshop* (U. Banerjee *et al.*, eds.), Lecture Notes in Computer Science, Springer-Verlag, 1992, pages 18–34.
3. “Reducing Synchronization Overhead for Compiler-Parallelized Codes on Software DSMs (Extended Abstract),” H. Han, C.-W. Tseng, and P. Keleher, *Languages and Compilers for Parallel Computing, Tenth International Workshop* (Z. Li *et al.*, eds.), Lecture Notes in Computer Science, Springer-Verlag, 1997, pages 240–245.
4. “Improving Compiler and Run-Time Support for Irregular Reductions Using Local Writes,” H. Han and C.-W. Tseng, *Languages and Compilers for Parallel Computing, Eleventh International Workshop* (S. Chatterjee *et al.*, eds.), Lecture Notes in Computer Science, Springer-Verlag, 1998, pages 181–196.
5. “Compiler and Run-time Support for Improving Locality in Scientific Codes (Extended Abstract),” H. Han, G. Rivera, and C.-W. Tseng, *Languages and Compilers for Parallel Computing, Twelfth International Workshop* (J. Ferrante *et al.*, eds.), Lecture Notes in Computer Science, Springer-Verlag, 1999, pages 455–458.
6. “A Comparison of Locality Transformations for Irregular Codes,” H. Han and C.-W. Tseng, *5th Workshop on Languages, Compilers, and Run-time Systems for Scalable Computers (LCR’2000)*, (S. Dwarkadas ed.), Lecture Notes in Computer Science, Springer-Verlag, 2000, pages 70–84.
7. “Searching Sequence Databases Using High Performance BLASTs,” X. Wu and C.-W. Tseng, *Parallel Computing for Bioinformatics and Computational Biology*, (A. Zomaya, ed.), John Wiley & Sons., February 2006.
8. “Peptide Identification via Tandem Mass Spectroscopy,” X. Wu, N. Edwards, and C.-W. Tseng, *Advances in Computers: Computational Biology and Bioinformatics*, Volume 68, (C.-W. Tseng, ed.), Elsevier, 2006.

1. "Interactive Parallel Programming Using the ParaScope Editor," K. Kennedy, K. McKinley, and C.-W. Tseng, *IEEE Transactions on Parallel and Distributed Systems*, July 1991, Vol. 2, No. 3, pages 329–341.
2. "Compiling Fortran D for MIMD Distributed-Memory Machines," S. Hiranandani, K. Kennedy, and C.-W. Tseng, *Communications of the ACM*, August 1992, Vol. 35, No. 8, pages 66–80 (invited paper from Supercomputing'91).
3. "The Power Test for Data Dependence," M. Wolfe and C.-W. Tseng, *IEEE Transactions on Parallel and Distributed Systems*, September 1992, Vol. 3, No. 5, pages 591–601.
4. "Analysis and Transformation in an Interactive Parallel Programming Tool," K. Kennedy, K. McKinley, and C.-W. Tseng, *Concurrency: Practice and Experience*, October 1993, Vol. 5, No. 7, pages 575–602.
5. "Unified Compilation of Fortran 77D and 90D," A. Choudhary, G. Fox, S. Hiranandani, K. Kennedy, C. Koelbel, S. Ranka, and C.-W. Tseng, *Letters on Programming Languages and Systems*, March–December 1993, Vol. 2, No. 1–4, pages 95–114.
6. "Evaluating Compiler Optimizations for Fortran D," S. Hiranandani, K. Kennedy, and C.-W. Tseng, *Journal of Parallel and Distributed Computing*, April 1994, Vol. 21, No. 1, pages 27–45.
7. "Requirements for Data-Parallel Programming Environments," V. Adve, A. Carle, E. Granston, S. Hiranandani, K. Kennedy, C. Koelbel, J. Mellor-Crummey, C.-W. Tseng, and S. Warren, *IEEE Transactions on Parallel and Distributed Technology*, Fall 1994, Vol. 2, No. 3, pages 48–58.
8. "Integrated Support for Task and Data Parallelism," K. M. Chandy, I. Foster, K. Kennedy, C. Koelbel, and C.-W. Tseng, *International Journal of Supercomputing Applications*, Summer 1994, Vol. 8, No. 2, pages 80–98.
9. "Improving Data Locality with Loop Transformations," K. McKinley, S. Carr, and C.-W. Tseng, *ACM Transactions on Programming Languages and Systems*, July 1996, Vol. 18, No. 4, pages 424–453.
10. "Memory Referencing Behavior in Compiler-Parallelized Applications," E. Torrie, M. Martonosi, M. Hall, and C.-W. Tseng, *International Journal of Parallel Programming*, August 1996, Vol. 24, No. 4 (invited paper from PACT'96).
11. "Interprocedural Compilation of Fortran D," M. Hall., S. Hiranandani, K. Kennedy, and C.-W. Tseng, *Journal of Parallel and Distributed Computing*, November 1996, Vol. 38, No. 2, pages 114–129.
12. "Characterizing the Memory Behavior of Compiler-Parallelized Applications," E. Torrie, M. Martonosi, C.-W. Tseng, and M. Hall, *IEEE Transactions on Parallel and Distributed Systems*, December 1996, Vol. 7, No. 12, pages 1225–1238.

13. “Performance of the Prototype Fortran D Compiler,” C.-W. Tseng, *Software—Practice and Experience*, July 1997, Vol. 7, No. 27, pages 763–796.
14. “Eliminating Barrier Synchronization for Compiler-Parallelized Codes on Software DSMs,” H. Han, C.-W. Tseng, and P. Keleher, *International Journal of Parallel Programming*, October 1998, Vol. 26, No. 5, pages 591–612 (invited paper from LCPC’97).
15. “Efficient Compiler and Run-Time Support for Parallel Irregular Reductions,” H. Han and C.-W. Tseng, *Parallel Computing*, 26(13-24):1861–1887, 2000 (special issue on Parallel Computing for Irregular Applications).
16. “Towards a First Vertical Prototyping of an Extremely Fine-grained Parallel Programming Approach,” D. Naishlos, J. Nuzman, C.-W. Tseng, and U. Vishkin, *Theory of Computing Systems*, August 2003, Vol. 36, pages 521–552 (invited paper from SPAA’01).
17. “The Efficacy of Software Prefetching and Locality Optimizations on Future Memory Systems,” A.-H. Badawy, A. Aggarwal, D. Yeung, and C.-W. Tseng, *Journal of Instruction-Level Parallelism*, October 2004, Vol. 6.
18. “Exploiting Locality for Irregular Scientific Codes,” H. Han and C.-W. Tseng, *IEEE Transactions on Parallel and Distributed Systems*, 17(7):606-618, July 2006.
19. “HMMatch: Peptide Identification by Spectral Matching of Tandem Mass Spectra Using Hidden Markov Models,” X. Wu, C.-W. Tseng, and N. Edwards, *Journal of Computational Biology*, 14(8):1025-1043, 2007.
20. “A Message Passing Benchmark for Unbalanced Applications,” J. Dinan, S. Olivier, G. Sabin, J. Prins, P. Sadayappan, and C.-W. Tseng, *Simulation Modelling Practice and Theory*, 16(9): 1177-1189, 2008.
21. “An Unsupervised, Model-Free, Machine-Learning Combiner for Peptide Identification from Tandem Mass Spectra,” N. Edwards, X. Wu. and C.-W. Tseng, *Clinical Proteomics*, 5(1):23-36, March 2009.

b¹ REFEREED CONFERENCE PROCEEDINGS

1. “Analysis and Transformation in the ParaScope Editor,” K. Kennedy, K. McKinley, and C.-W. Tseng, *ACM 1991 International Conference on Supercomputing (ICS’91)*, Cologne, Germany, June 1991, pages 433–447.
2. “Practical Dependence Testing,” G. Goff, K. Kennedy, and C.-W. Tseng, *ACM SIGPLAN’91 Conference on Programming Language Design and Implementation (PLDI’91)*, Toronto, Canada, June 1991, pages 15–29.
3. “Compiler Optimizations for Fortran D on MIMD Distributed-Memory Machines,” S. Hiranandani, K. Kennedy, and C.-W. Tseng, *Supercomputing’91*, Albuquerque, NM, November 1991, pages 86–100.

4. "Evaluation of Compiler Optimizations for Fortran D on MIMD Distributed-Memory Machines," S. Hiranandani, K. Kennedy, and C.-W. Tseng, *ACM 1992 International Conference on Supercomputing (ICS'92)*, Washington DC, July 1992, pages 1–14.
5. "Compiling Fortran 77D and 90D for MIMD Distributed-Memory Machines," A. Choudhary, G. Fox, S. Hiranandani, K. Kennedy, C. Koelbel, S. Ranka, and C.-W. Tseng, *4th Symposium on the Frontiers of Massively Parallel Computation (Frontiers'92)*, McLean, VA, October 1992, pages 4–11.
6. "Interprocedural Compilation of Fortran D for MIMD Distributed-Memory Machines," M. Hall, S. Hiranandani, K. Kennedy, and C.-W. Tseng, *Supercomputing'92*, Minneapolis, MN, November 1992, pages 522–535.
7. "Preliminary Experiences with the Fortran D Compiler," S. Hiranandani, K. Kennedy, and C.-W. Tseng, *Supercomputing'93*, Portland, OR, November 1993, pages 338–350.
8. "Compiler Optimizations for Improving Data Locality," S. Carr, K. McKinley, and C.-W. Tseng, *Sixth International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS-VI)*, San Jose, CA, October 1994, pages 252–262.
9. "The D Editor: A New Interactive Parallel Programming Tool," S. Hiranandani, K. Kennedy, C.-W. Tseng, and S. Warren, *Supercomputing'94*, Washington, DC, November 1994, pages 733–742.
10. "An Overview of the SUIF Compiler for Scalable Parallel Machines," S. Amarasinghe, J. Anderson, M. Lam, and C.-W. Tseng, *Seventh SIAM Conference on Parallel Processing for Scientific Computing (SIAM'95)*, San Francisco, CA, February 1995, pages 1–7.
11. "Evaluating the Impact of Advanced Memory Systems on Compiler-Parallelized Codes," E. Torrie, C.-W. Tseng, M. Martonosi, and M. Hall, *International Conference on Parallel Architectures and Compilation Techniques (PACT'95)*, Limassol, Cyprus, June 1995, pages 1–10.
12. "Unified Compilation Techniques for Shared and Distributed Address Space Machines," C.-W. Tseng, J. Anderson, S. Amarasinghe, and M. Lam, *ACM International Conference on Supercomputing (ICS'95)*, Barcelona, Spain, July 1995, pages 67–76.
13. "Compiler Optimizations for Eliminating Barrier Synchronization," C.-W. Tseng, *ACM Symposium on Principles & Practice of Parallel Programming (PPOPP'95)*, Santa Barbara, CA, July 1995, pages 144–155.
14. "Enhancing Software DSM for Compiler-Parallelized Applications," P. Keleher and C.-W. Tseng, *11th International Parallel Processing Symposium (IPPS'97)*, Geneva, Switzerland, April 1997, pages 490–499 (extended paper).
15. "Compile-time Synchronization Optimizations for Software DSMs," H. Han and C.-W. Tseng, *12th International Parallel Processing Symposium (IPPS'98)*, Orlando, FL, April 1998, pages 662–669 (extended paper).
16. "Data Transformations for Eliminating Conflict Misses," G. Rivera and C.-W. Tseng, *ACM SIGPLAN'98 Conference on Programming Language Design and Implementation (PLDI'98)*, Montreal, Canada, June 1998, pages 38–49.

17. "Eliminating Conflict Misses for High Performance Architectures," G. Rivera and C.-W.⁶ Tseng, *ACM International Conference on Supercomputing (ICS'98)*, Melbourne, Australia, July 1998, pages 353–360.
18. "Improving Compiler and Run-Time Support for Adaptive Irregular Codes," H. Han and C.-W. Tseng, *International Conference on Parallel Architectures and Compilation Techniques (PACT'98)*, Paris, France, October 1998, pages 1–8.
19. "A Comparison of Compiler Tiling Algorithms," G. Rivera and C.-W. Tseng, *8th International Conference on Compiler Construction (CC'99)*, Amsterdam, The Netherlands, March 1999, pages 168–182.
20. "Locality Optimizations for Multi-Level Caches," G. Rivera and C.-W. Tseng, *SC'99*, Portland, OR, November 1999, pages 1–16.
21. "Tiling Optimizations for 3D Scientific Computations," G. Rivera and C.-W. Tseng, *SC'00*, Dallas, TX, November 2000, pages 1–23.
22. "A Comparison of Parallelization Techniques for Irregular Reductions," H. Han and C.-W. Tseng, *International Parallel and Distributed Processing Symposium (IPDPS'01)*, San Francisco CA, April 2001 (extended paper).
23. "Evaluating the Impact of Memory System Performance on Software Prefetching and Locality Optimizations," A.-H. Badawy, A. Aggarwal, D. Yeung, and C.-W. Tseng, *ACM International Conference on Supercomputing (ICS'01)*, Sorrento, Italy, June 2001.
24. "Towards a First Vertical Prototyping of an Extremely Fine-grained Parallel Programming Approach," D. Naishlos, J. Nuzman, C.-W. Tseng, and U. Vishkin, *13th ACM Symposium on Parallel Algorithms and Architectures (SPAA'01)*, Crete, Greece, July 2001.
25. "BioBench: A Benchmark Suite of Bioinformatics Applications," K. Albayraktaroglu, A. Jaleel, X. Wu, B. Jacob, M. Franklin, C.-W. Tseng, and D. Yeung, *IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS)*, Austin, TX, March 2005.

b² REFEREED WORKSHOPS

1. "Design and Implementation of the D Editor," S. Hiranandani, K. Kennedy, C.-W. Tseng, and S. Warren, *Proceedings of the Second Workshop on Environments and Tools for Parallel Scientific Computing, SIAM*, Townsend, TN, May 1994, pages 1–10.
2. "Communication Analysis for Shared and Distributed Memory Machines," C.-W. Tseng, *Workshop on Compiler Optimizations on Distributed Memory Systems*, 7th IEEE Symposium on Parallel and Distributed Processing, San Antonio, TX, October 1995, pages 1–10.
3. "Improving the Compiler/Software DSM Interface: Preliminary Experiences," P. Keleher and C.-W. Tseng, *First SUIF Compiler Workshop*, Stanford University, Stanford, CA, January 1996, pages 1–17.

4. “Reducing Synchronization Overhead for Compiler-Parallelized Codes on Software DSMs,”⁷ H. Han, C.-W. Tseng, and P. Keleher, *Workshop on Languages and Compilers for Parallel Computing (LCPC’97)*, Minneapolis, MN, August 1997, pages 240–245.
5. “Compiler Optimizations for High Performance Architectures,” H. Han, G. Rivera, and C.-W. Tseng, *Second SUIF Compiler Workshop*, Stanford University, Stanford, CA, August 1997, pages 1–11.
6. “Improving Parallelizing Compiler Support for Irregular Reductions,” H. Han, and C.-W. Tseng, *Workshop on Languages and Compilers for Parallel Computing (LCPC’98)*, Chapel Hill, NC, August 1998, pages 175–190.
7. “Eliminating Conflict Misses for Tiled Codes,” G. Rivera, and C.-W. Tseng, *Dagstuhl Seminar on Tiling for Optimal Resource Utilization*, Wadern, Germany, August 1998, pages 16–17.
8. “Software Support for Improving Locality in Scientific Codes,” H. Han, G. Rivera, and C.-W. Tseng, *8th Workshop on Compilers for Parallel Computers (CPC’2000)*, Aussois, France, January 2000, pages 213–228.
9. “A Comparison of Locality Transformations for Irregular Codes,” H. Han and C.-W. Tseng, *5th Workshop on Languages, Compilers, and Run-time Systems for Scalable Computers (LCR’2000)*, Rochester, NY, May 2000, pages 31–36.
10. “Improving Locality For Adaptive Irregular Scientific Codes,” H. Han and C.-W. Tseng, *Workshop on Languages and Compilers for High-Performance Computing (LCPC’00)*, White Plains, NY, August 2000, pages 1–15.
11. “Evaluating Multi-threading in the Prototype XMT Environment,” D. Naishlos, J. Nuzman, C.-W. Tseng, and U. Vishkin, *4th ACM Workshop on Multi-threaded Execution, Architecture, and Compilation (MTEAC-4)*, Monterey, CA, December 2000, pages 1–8 (best paper award).
12. “Evaluating the XMT Parallel Programming Model,” D. Naishlos, J. Nuzman, C.-W. Tseng, and U. Vishkin, *Workshop on High-level Programming Models and Supportive Environments (HIPS’01)*, San Francisco, CA, April 2001.
13. “Evaluating the Impact of Programming Language Features on the Performance of Parallel Applications on Cluster Architectures,” K. Berlin, J. Huan, M. Jacob, G. Kochhar, J. Prins, B. Pugh, P. Sadayappan, J. Spacco, and C.-W. Tseng, *Workshop on Languages and Compilers for High-Performance Computing (LCPC’03)*, College Station, TX, October 2003.
14. “ESTmapper: Efficiently Aligning DNA Sequences to Genomes,” X. Wu, W.-J. Lee, and C.-W. Tseng, *4th IEEE International Workshop on High Performance Computational Biology (HiCOMB)*, Denver, CO, April 2005.
15. “UTS: An Unbalanced Tree Search Benchmark,” S. Olivier, J. Huan, J. Liu, J. Prins, J. Dinan, P. Sadayappan and C.-W. Tseng, *Workshop on Languages and Compilers for High-Performance Computing (LCPC’06)*, New Orleans, LA, November 2006.
16. “Dynamic Load Balancing of Unbalanced Computations Using Message Passing,” J. Dinan, S. Olivier, J. Huan, J. Liu, J. Prins, P. Sadayappan and C.-W. Tseng *6th International Workshop on Performance Modeling, Evaluation, and Optimization of Parallel and Distributed Systems (PMEO-PDS 2007)*, Long Beach, CA, March 2007.

1. “Fortran D Language Specification,” G. Fox, S. Hiranandani, K. Kennedy, C. Koelbel, U. Kremer, C.-W. Tseng, and M. Wu, Computer Science Technical Report 90-141, Rice University, December 1990.
2. “An Optimizing Fortran D Compiler for MIMD Distributed-Memory Machines,” C.-W. Tseng, Ph.D. Thesis, Computer Science Technical Report 93-199, Rice University, January 1993.
3. “SUIF: An Infrastructure For Research on Parallelizing and Optimizing Compilers,” R. Wilson, R. French, C. Wilson, S. Amarasinghe, J. Anderson, S. Tjiang, S.-W. Liao, C.-W. Tseng, M. Hall, M. Lam, and J. Hennessy, SIGPLAN Notices 29:31–37, 1994.
4. “SUIF: A Parallelizing and Optimizing Research Compiler,” R. Wilson, R. French, C. Wilson, S. Amarasinghe, J. Anderson, S. Tjiang, S.-W. Liao, C.-W. Tseng, M. Hall, M. Lam, and J. Hennessy, Technical Report CSL-TR-94-620, Computer Systems Laboratory, Stanford University, May 1994.
5. “Efficient Machine-Independent Programming of High-Performance Multiprocessors,” C.-W. Tseng, Technical Report CS-TR-3545, Dept. of Computer Science, University of Maryland, October 1995.
6. “Data Layout Optimizations for High-Performance Architectures,” C.-W. Tseng, Technical Report CS-TR-3818, Dept. of Computer Science, University of Maryland, February 1997.

e TALKS, ABSTRACTS, AND OTHER PAPERS PRESENTED

i INVITED TALKS

- “Compilation Support for High-Performance Multiprocessors,” C.-W. Tseng, Dept. of Computer Science, University of Massachusetts at Amherst, June 5, 1996.
- “Compilation Support for Scalable Multiprocessors,” C.-W. Tseng, Dept. of Computer Science, University of Delaware, October 18, 1996.
- “Is Compiler Optimization Research Still Relevant?” C.-W. Tseng, Keynote Address, *3rd Workshop on Compiler Techniques for High Performance Computing (CTHPC 2003)*, March 21, 2003.

i TALKS

- “Improving the Reliability of Peptide Identifications using Machine Learning,” X. Wu, C.-W. Tseng, and N. Edwards, 4th Annual Conference of the US Human Proteome Organisation (US HUPO 2008), Bethesda, MD, March 2008.

- “ESTmapper: Efficiently Clustering EST Sequences Using Genome Maps,” X. Wu, W.-J. Lee, D. Gupta, and C.-W. Tseng, 8th International Conference on Research in Computational Molecular Biology (RECOMB 2004), San Diego, CA, March 2004.
- “Searching Bioinformatic Sequence Databases using UM-BLAST—A Wrapper for High-Performance BLASTs,” X. Wu and C.-W. Tseng, 8th International Conference on Research in Computational Molecular Biology (RECOMB 2004), San Diego, CA, March 2004.
- “Novel Peptide Identification Using ESTs and Genomic Sequence,” N. Edwards, X. Wu, and C.-W. Tseng, 2nd Annual Conference of the US Human Proteome Organisation (US HUPO 2006), Boston, MA, March 2006.
- “Experimental Comparison of Peptide Identification Algorithms,” X. Wu, N. Edwards, and C.-W. Tseng, 9th Annual Conference on Computational Genomics, Baltimore, MD, October 2006.
- “Peptide Identification by Spectral Matching of Tandem Mass Spectra using Hidden Markov Models,” X. Wu, N. Edwards, and C.-W. Tseng, 2006 RECOMB Satellite Conferences on Systems Biology and Computational Proteomics, San Diego, CA, December 2006.
- “Peptide Identification by Spectral Matching of Tandem Mass Spectra using Hidden Markov Models,” X. Wu, C.-W. Tseng, and N. Edwards, 4th Annual Conference of the US Human Proteome Organisation (US HUPO 2008), Bethesda, MD, March 2008.

i CONTRACTS AND GRANTS

- NSF CISE Postdoctoral Research Associate in Experimental Science, *National Science Foundation*, September 1993–August 1995, \$88,000.
- “Compiler Support for High-Performance Multiprocessors,” *General Research Board of Graduate Studies and Research, University of Maryland*, Principal Investigator, June–September 1996, \$6,250.
- “Efficient Machine-Independent Programming of High-Performance Multiprocessors,” *National Science Foundation*, Principal Investigator, June 1996–May 2000, \$200,000.
- “Data Layout Optimizations for High Performance Architectures,” *National Science Foundation*, Principal Investigator, September 1997–August 2000, \$186,000.
- “Uncovering and Exploiting Memory Parallelism in Pointer-Chasing Applications,” *National Science Foundation*, co-Principal Investigator (with D. Yeung (PI)), July 2000–June 2003, \$320,000.
- “System Support for Enterprise Application Servers,” *National Science Foundation*, Senior Personnel (with P. Keleher (PI) *et al.*), September 2000–September 2003, \$861,244.

- “Parallel Programming Paradigms for Distributed Memory and Distributed Shared Memory High Performance Computers,” *National Security Agency*, Principal Investigator (with B. Pugh (PI), S. Chatterjee (co-PI), and P. Sadayappan (co-PI)), November 2000–October 2001, \$320,000.
- “Software Support For Improving Locality in Advanced Scientific Codes,” *National Science Foundation*, Principal Investigator, September 2001–August 2004, \$215,000.
- “Parallel Programming Paradigms for Distributed Memory and Distributed Shared Memory High Performance Computers (2nd Year),” *National Security Agency*, Principal Investigator (with B. Pugh (PI), J. Prins (co-PI), and P. Sadayappan (co-PI)), April 2002–March 2003, \$345,000.
- “ITR: Parallel Random-Access Model (PRAM)-on-a-Chip,” *National Science Foundation*, Senior Personnel (with U. Vishkin (PI) *et al.*), September 2003–August 2008, \$750,000.
- “Unbalanced Tree Search Benchmark,” *National Security Agency*, Principal Investigator (with J. Prins and P. Sadayappan), May 2004–April 2006, \$500,000.
- “Proteomic Characterization of Alternate Splicing and cSNP Protein Isoforms,” *HHS-PHS/National Institute of Health*, Co-PI (with N. Edwards (PI), C. Fenselau (co-PI)), September 2006–August 2009, \$853,442.
- “Tandem Mass Spectral Biomarkers for Plant Pathogens,” *USDA Cooperative Agreement*, transfer from N. Edwards, February 2007–June 2008, \$46,524.

j FELLOWSHIPS, PRIZES, AND AWARDS

- DARPA Graduate Fellow in Parallel Processing (awarded).
- Budd Award for Best Thesis in School of Engineering, Rice University, 1993.
- NSF CISE Postdoctoral Research Associate in Experimental Science, 1993.
- NSF CAREER Development Award, 1996.

k EDITORSHIPS, EDITORIAL BOARDS, AND REVIEWING ACTIVITIES

i REVIEWING ACTIVITIES FOR JOURNALS

- *ACM Transactions on Programming Languages and Systems (TOPLAS)*
- *ACM Letters on Programming Languages and Systems (LOPLAS)*
- *Analytical Chemistry*
- *Concurrency: Practice and Experience (CP&E)*
- *IEEE Computer*

- *IEEE Transactions on Computer Systems (TOCS)*
- *IEEE Transactions on Parallel and Distributed Systems (TPDS)*
- *IEEE Transactions on Software Engineering (TSE)*
- *International Journal of High Speed Computing (IJHSC)*
- *International Journal of Parallel Programming (IJPP)*
- *Journal of Parallel and Distributed Computing (JPDC)*
- *Journal of Programming Languages (JPL)*
- *Journal of Supercomputing (JOS)*
- *Parallel Computing*
- *Parallel and Distributed Computing Practices*
- *Scientific Computing*
- *Scientific Programming*
- *Software: Practice & Experience*
- *Transactions on Architecture and Code Optimization (TACO)*

ii **REVIEWING ACTIVITIES FOR REFEREED CONFERENCES**

- *International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*
- *International Symposium on Compiler Construction (CC)*
- *Workshop on Compilers and Operating Systems for Low Power (COLP)*
- *European Conference on Parallel Processing (EURO-PAR)*
- *Symposium on the Frontiers of Massively Parallel Computing (Frontiers)*
- *Workshop on High-Level Parallel Programming Models and Supportive Environments (HIPS)*
- *High Performance Computer Architecture (HPCA)*
- *International Conference on High Performance Parallel Computing (HiPC)*
- *International Conference on Computer Languages (ICCL)*
- *International Conference on Distributed Computing Systems (ICDCS)*
- *International Conference on Parallel Processing (ICPP)*
- *International Conference on Supercomputing (ICS)*

- *International Parallel Processing Symposium (IPPS)*
- *International Symposium on Computer Architecture (ISCA)*
- *International Symposium on High Performance Computing (ISHPC)*
- *International Symposium on Performance Analysis of Systems and Software (ISPASS)*
- *Workshop on Languages and Compilers for Parallel Computing (LCPC)*
- *Workshop on Languages, Compilers, and Run-time Systems for Scalable Computers (LCR)*
- *Object-Oriented Programming Systems, Languages, and Applications (OOPSLA)*
- *Symposium on Operating Systems Design and Implementation (OSDI)*
- *Parallel Architectures and Compilation Techniques (PACT)*
- *Conference on Programming Language Design and Implementation (PLDI)*
- *IASTED Conference on Parallel and Distributed Computing and Systems (PDCS)*
- *Conference on Principles of Distributed Computing (PODC)*
- *Conference on Principles of Programming Languages (POPL)*
- *Conference on Principles and Practice of Parallel Programming (PPOPP)*
- *Scalable High Performance Computing Conference (SHPCC)*
- *Conference on Measurement and Modeling of Computer Systems (SIGMETRICS)*
- *ACM Conference on Supercomputing (SC)*

1 RESEARCH SOFTWARE

- *ParaScope Dependence Analyzer*, 1989–91, (with P. Havlak)
Incrementally computes data dependences for scientific Fortran programs.
- *ParaScope Editor*, 1990–93, (with K. McKinley *et al.*)
Programming environment geared towards developing parallel Fortran programs for shared-memory multiprocessors. Provides interactive analysis and program transformations.
- *Fortran D Compiler*, 1991–94, (with S. Hiranandani)
Compiles a data-parallel version of Fortran which provides data decomposition specifications. The compiler produces message-passing codes for *multiple-instruction, multiple-data* (MIMD) distributed-memory machines such as the IBM SP-2.
- *D Editor*, 1993–94, (with S. Hiranandani and S. Warren)
Programming environment for mapping communication and performance information back to the Fortran D program.

- *SUIF Parallelizing Compiler*, 1993–95 (with S. Amarasinghe, J. Anderson *et al.*)
Parallelizing compiler for shared-memory multiprocessors. Investigated approaches for combining capabilities of compilers for both shared and distributed-memory multiprocessors.
- *COSMIC*, 1995–2001 (with H. Han and G. Rivera)
Prototype compiler based on the SUIF compiler targeting software distributed-shared-memory (DSM) systems and data layout optimizations on advanced microprocessors.
- *ESTmapper*, 2004–2005 (with X. Wu)
Prototype DNA to genome alignment tool based on WOTD suffix trees. Useful for EST clustering, gene finding, and identifying gene regulation signals. Efficiency enables processing large data sets. Parallel implementations for both shared-memory multiprocessors and PC clusters.
- *HMMatch*, 2006–2007 (with X. Wu and N. Edwards)
Peptide identification by tandem mass spectrometry typically relies on sequence-based search engines that compare experimental spectra to theoretical spectra. In other fields, spectral matching is used to compare new spectra with libraries of identified spectra. HMMatch is Hidden Markov Model-based method to spectral matching, in which many examples of a peptide's fragmentation spectrum are summarized in a generative probabilistic model that captures not only the expected ion intensities, but also the variation in the intensities of peaks.
- *PepArML*, 2007–2008 (with X. Wu and N. Edwards)
PepArML (Peptide identification Arbiter by Machine Learning) is an unsupervised, model-free framework that combines results from multiple tandem mass spectra database search engines to improve the precision of peptide identifications. PepArML is based on the algorithms implemented in the Weka machine learning toolkit. The PepArML predictor is trained on the fly for each new set of search results without user intervention, making it robust for different instruments, search engines, and search engine parameters.

a COURSES

ii SPECIALIZED COURSES

| Semester | Course | # Students | Description |
|--------------|---------------|------------|--|
| Fall, 1995 | CMSC 430 | 37 | Theory of Language Translation |
| Spring, 1996 | CMSC 731/838T | 16 | Programming Language Implementation |
| Fall, 1996 | CMSC 732/838T | 10 | Compiling for High Performance Architectures |
| Spring, 1997 | CMSC 430 | 55 | Theory of Language Translation |
| Fall, 1997 | CMSC 430 | 24 | Theory of Language Translation |
| Spring, 1998 | CMSC 731/838T | 21 | Programming Language Implementation |
| Fall, 1998 | CMSC 430 | 35 | Theory of Language Translation |
| Spring, 1999 | CMSC 732/838T | 8 | Compiling for High Performance Architectures |
| Spring, 2000 | CMSC 430 | 24 | Theory of Language Translation |
| Fall, 2000 | CMSC 430 | 53 | Theory of Language Translation |
| Spring, 2001 | CMSC 430 | 43 | Theory of Language Translation |
| Fall, 2002 | CMSC 430 | 36 | Theory of Language Translation |
| Spring, 2003 | CMSC 838T | 19 | High Performance Computing & Bioinformatics |
| Fall, 2003 | CMSC 430 | 16 | Theory of Language Translation |
| Spring, 2004 | CMSC 838T | 13 | Bioinformatics & High Performance Computing |
| Spring, 2005 | CMSC 132 | 41 | Object-Oriented Programming II |
| Fall, 2005 | CMSC 132 | 40 | Object-Oriented Programming II |
| Spring, 2006 | CMSC 430 | 15 | Theory of Language Translation |
| Fall, 2006 | CMSC 132H | 16 | Object-Oriented Programming II (Honors) |
| Spring, 2007 | CMSC 132H | 11 | Object-Oriented Programming II (Honors) |
| Fall, 2007 | CMSC 132H | 20 | Object-Oriented Programming II (Honors) |
| Spring, 2008 | CMSC 330 | 87 | Organization of Programming Languages |
| Spring, 2009 | CMSC 330 | 47 | Organization of Programming Languages |
| Spring, 2009 | CMSC 430 | 17 | Theory of Language Translation |
| Fall, 2009 | CMSC 330 | 50 | Organization of Programming Languages |
| Fall, 2009 | CMSC 411 | 14 | Computer Systems Architecture |
| Spring, 2010 | CMSC 330 | 65 | Organization of Programming Languages |
| Fall, 2010 | CMSC 330 | 109 | Organization of Programming Languages |
| Spring, 2011 | CMSC 411 | 29 | Computer Systems Architecture |
| Spring, 2011 | CMSC 430 | 17 | Introduction to Compilers |
| Fall, 2011 | CMSC 330 | 122 | Organization of Programming Languages |
| Spring, 2012 | CMSC 330 | 174 | Organization of Programming Languages |
| Spring, 2012 | CMSC 411 | 30 | Computer Systems Architecture |
| Fall, 2013 | CMSC 330 | 205 | Organization of Programming Languages |
| Fall, 2013 | CMSC 411 | 38 | Computer Systems Architecture |
| Spring, 2014 | CMSC 330 | 198 | Organization of Programming Languages |
| Spring, 2014 | CMSC 430 | 49 | Introduction to Compilers |

| Semester | Course | # Students | Description |
|--------------|----------|------------|-----------------|
| Summer, 1996 | CMSC 386 | 1 | Exper. Learning |
| Fall, 1996 | CMSC 386 | 1 | Exper. Learning |
| Spring, 1998 | CMSC 386 | 1 | Exper. Learning |
| Summer, 1998 | CMSC 386 | 1 | Exper. Learning |
| Summer, 2004 | CMSC 390 | 1 | Honors Paper |

b COURSES OR CURRICULUM DEVELOPMENT

CMSC430 – Theory of Language Translation

- Updated course by extending syllabus to include material on Java byte code, global register allocation, instruction scheduling, compiling functional and object-oriented languages, as well as an introduction to locality and parallelism optimizations. Improvements provide students with awareness of current issues in compilation.
- Developed new programming project to build a C compiler which generates Java byte code. Wrote 5000+ lines of code in Java as a base for students. Students test their compiler by running their output on the Java Virtual Machine (JVM) on multiple architectures.

CMSC731 – Programming Language Implementation

- Redeveloped entire course and projects as an introductory graduate level compiler course. Course focuses on compilation techniques needed to obtain good performance on advanced microprocessors. Syllabus includes data-flow and interprocedural analyses, classical compiler optimizations, code generation techniques, and an introduction to dependence analysis and loop transformations. The course provide students with awareness of techniques required in modern commercial compilers.
- Created a programming project to build an optimizing compiler based on the Stanford SUIF compiler. Students implement local optimizations, a framework for iterative global data-flow analysis, and several classical compiler optimizations such as partial redundancy elimination. Students also conduct experiments to evaluate the efficacy of their optimizations on important computation kernels and standard benchmark suites.

CMSC732 – Compiling for High Performance Architectures

- Developed entire course as an advanced graduate level course on optimizing compilers. Course focuses on compilation techniques needed to obtain high performance on advanced parallel architectures. Syllabus includes dependence analysis, loop transformations, parallelization techniques, memory hierarchy management, programming environments, and performance

debugging. The course provide students with awareness of current research issues in compilation. 16

- Supervised students in individual research projects on compilation techniques. Projects include implementing and evaluating selected optimizations in the Stanford SUIF compiler. Several projects resulted in conference and journal publications.

CMSC838T – High Performance Computing & Bioinformatics

- Developed an advanced graduate level course focusing on bioinformatics applications, high-performance computing, and the application of high-performance computing to bioinformatics applications. Bioinformatics is the creation and development of advanced information and computational techniques for problems in biology, especially methods for storing, retrieving, analyzing, classifying, and predicting biological functions from nucleic acid (DNA/RNA) and protein (amino acid) sequence data. High-performance computing describes a set of hardware and software techniques developed for building computer systems capable of quickly performing large amounts of computation.
- Taught students in course to 1) learn about characteristics of bioinformatic applications, 2) examine software techniques used in high-performance computing, and 3) study how to apply high-performance computing to bioinformatic applications.

e ADVISING: NON-RESEARCH DIRECTION

ii GRADUATE (PH.D. COMMITTEE)

Wayne Kelly, 1997 (Bill Pugh, advisor).
Evan Rosser, 1998 (Bill Pugh, advisor).
Yuan-Shin Hwang, 1999 (Joel Saltz, advisor).
Tatiana Shpeisman, 1999 (Bill Pugh, advisor).
Mustafa Uysal, 1999 (Joel Saltz, advisor).
Sean Luke, 2000 (Jim Hendler, advisor).
Ugur Cetintemel, 2001 (Pete Keleher, advisor).
Renato Ferreira, 2001 (Joel Saltz, advisor).
Jung-Min Kim, 2001 (Adam Porter, advisor).
Henrique Andrade, 2002 (Joel Saltz, advisor).
Anasua Bhowmik, 2003 (Manoj Franklin, advisor).
Mohamed Zahran, 2003 (Manoj Franklin, advisor).
Aneesh Aggarwal, 2003 (Manoj Franklin, advisor).
Zhexuan Song, 2003 (Nick Roussopoulos, advisor).
Bryan Buck, 2004 (Jeff Hollingsworth, advisor).
Shaoxiong Hua, 2004 (Gang Qu, advisor).
Dongkeun Kim, 2004 (Don Yeung, advisor).

Steven Haga, 2005 (Rajeev Barua, advisor).
 Yongpan Yan, 2005 (John Moulton, advisor).
 Aamer Jaleel, 2005 (Bruce Jacob, advisor).
 Seungryul Choi, 2006 (Don Yeung, advisor).
 Mingyung Ko, 2006 (Shuvra Bhattacharyya, advisor).
 Dong-Ik Ko, 2006 (Shuvra Bhattacharyya, advisor).
 Sumesh Udayakumar, 2006 (Shuvra Bhattacharyya, advisor).
 Samuel Rodriguez, 2006 (Bruce Jacob, advisor).
 Bao Trinh, 2006 (Ashok Agrawala, advisor).
 Angelo Dominguez, 2007 (Rajeev Barua, advisor).
 Kursad Albayraktaroglu, 2007 (Manoj Franklin, advisor).
 Chia-Jui (Jerry) Hsu, 2007 (Shuvra Bhattacharyya, advisor).
 Scott Thomas, 2007 (Don Perlis, advisor).
 Rezarta Islamaj, 2007 (Lise Getoor, advisor).
 Sadagopan Srinivasan, 2007 (Bruce Jacob, advisor).
 Shang-Chieh Wu, 2008 (Alan Sussman, advisor).
 Xiangrong Zhou, 2008 (Peter Petrov, advisor).
 Xun Yuan, 2008 (Atif Memon, advisor).
 Konstantin Berlin, 2010 (Dianne O'Leary, advisor).
 Matthew Simpson, 2010 (Rajeev Barua, advisor).
 Chenjie Yu, 2010 (Peter Petrov, advisor).
 George Caragea, 2011 (Uzi Vishkin, advisor).
 Meng-Ju Wu, 2012 (Don Yeung, advisor).

f **ADVISING: RESEARCH DIRECTION**

i **UNDERGRADUATE HONORS**

- Konstantin Berlin, December 2002.
- Shirin Mehraban, May 2004.

ii **MASTER'S**

- Zhengyu Wang, May 1997.
- Yanqing Zeng, May 2000.
- Dorit Naishlos, December 2000.
- Xue Wu, May 2002.

iii DOCTORAL

- Hwansoo Han, February 2001.
- Gabriel Rivera, February 2001.
- Xue Wu, August 2008.

4 SERVICE

a PROFESSIONAL

ii Unpaid Reviewing Activities For Agencies

- NSF panel, MIPS, 1994.
- NSF panel, CCR, 1995.
- NSF panel, ASC, 1996.
- NSF panel, Professional Opportunities for Women in Research and Education (POWRE), 1997.
- NSF panel, Small Business Innovation Research (SBIR), 1999.
- NSF panel, Graduate Research Fellowship Program (GRFP), 2000.
- NSF panel, CCR, 2002.
- NSF panel, ACR Software, 2003.
- NIH/NIAID panel, Genomics of Transplantation , 2004.
- NSF panel, Small Business Innovation Research (SBIR), 2004.
- NSF panel, Graduate Research Fellowship Program (GRFP), 2005.
- NSF panel, Compilers and Runtime Systems (CRTS), 2005.
- NSF panel, Graduate Research Fellowship Program (GRFP), 2006.
- NIH panel, High-Performance Computing Resources in Biology, 2006.
- NSF panel, Small Business Innovation Research (SBIR), 2006.
- NIH panel, Shared Instrumentation Program , 2006.
- NSF panel, Graduate Research Fellowship Program (GRFP), 2007.
- NSF panel, CAREER Genomics , 2007.

- NSF panel, Computer Research Infrastructure (CRI) , 2007.
- NSF panel, Graduate Research Fellowship Program (GRFP), 2008.
- NSF panel, BIO-MED CAREER , 2008.
- NSF panel, Graduate Research Fellowship Program (GRFP), 2009.

iv Other Non-University Committees

- Program Committee, *2nd SUIF Compiler Workshop*, August 1997.
- Program Committee, *12th International Parallel Processing Symposium (IPPS'98)*, April 1998.
- Program Committee, *8th International Conference on Compiler Construction (CC'99)*, March 1999.
- Program Committee, *ACM SIGPLAN'99 Conference on Programming Language Design and Implementation (PLDI'99)*, May 1999.
- Program Committee, *7th ACM SIGPLAN Symposium on Principles & Practice of Parallel Programming (PPOPP'99)*, May 1999.
- Program Committee, *2nd International Symposium on High Performance Computing (ISHPC'99)*, May 1999.
- Program Committee, *9th International Conference on Compiler Construction (CC'00)*, March 2000.
- Program Committee, *3rd International Symposium on High Performance Computing (ISHPC'2K)*, May 2000.
- Program Committee, *13th Workshop on Languages and Compilers for High Performance Computing (LCPC'00)*, August 2000.
- Program Committee, *1st Workshop on Compilers and Operating Systems for Low Power (COLP'00)*, October 2000.
- Program Committee, *14th International Parallel and Distributed Processing Symposium (IPDPS'01)*, April 2001.
- Program Committee, *14th Workshop on Languages and Compilers for High Performance Computing (LCPC'01)*, August 2001.
- Program Committee, *2nd Workshop on Compilers and Operating Systems for Low Power (COLP'01)*, October 2001.
- Program Co-chair, *6th Workshop on Languages, Compilers, and Run-time Systems for Scalable Computers (LCR'02)*, March 2002.

- Program Co-chair, *15th Workshop on Languages and Compilers for High-Performance Computing (LCPC'02)*, August 2002.
- Program Committee, *3rd Workshop on Compilers and Operating Systems for Low Power (COLP'02)*, September 2002.
- Program Committee, *European Conference on Parallel and Distributed Computing (Euro-Par'03)*, August 2003.
- Program Committee, *3rd Workshop on Compilers and Operating Systems for Low Power (COLP'03)*, October 2003.
- Program Committee, *16th Workshop on Languages and Compilers for High Performance Computing (LCPC'03)*, October 2003.
- Program Committee, *17th Workshop on Languages and Compilers for High Performance Computing (LCPC'04)*, September 2004.
- Program Committee, *7th Workshop on Languages, Compilers, and Run-time Systems for Scalable Computers (LCR'04)*, October 2004.
- Program Committee, *ACM SIGMICRO Conference on Computing Frontiers 2005 (CF'05)*, May 2005.
- Program Committee, *2005 International Conference on Parallel Processing (ICPP'05)*, June 2005.
- Program Committee, *17th IASTED Conference on Parallel and Distributed Computing and Systems (PDCS'05)*, November 2005.
- Program Chair, *5th IEEE International Workshop on High Performance Computational Biology (HiCOMB 2006)*, April 2006.
- Program Committee, *18th IASTED Conference on Parallel and Distributed Computing and Systems (PDCS'06)*, November 2006.
- Program Committee, *13th International Conference on High Performance Parallel Computing (HiPC'06)*, December 2006.
- Program Committee, *Workshop on Parallel Computational Biology (PBC'07)*, September 2007.
- Program Committee, *2007 International Conference on Parallel Processing (ICPP'07)*, September 2007.
- Program Committee, *19th IASTED Conference on Parallel and Distributed Computing and Systems (PDCS'07)*, November 2007.
- Program Committee, *8th IEEE International Workshop on High Performance Computational Biology (HiCOMB 2009)*, May 2009.
- Program Committee, *12th International Conference on Computational Science and Engineering (CSE-09)*, August 2009.

- Program Committee, *15th International Conference on Parallel and Distributed Systems (ICPADS'09)*, December 2009.
- Program Committee, *9th IEEE International Workshop on High Performance Computational Biology (HiCOMB 2010)*, April 2010.
- Program Committee, *2010 IEEE and ACM SIGARCH Conference on Supercomputing (SC10)*, November 2010.
- Program Committee, *13th International Conference on Computational Science and Engineering (CSE-2010)*, December 2010.
- Program Committee, *3rd International Conference on Bioinformatics and Computational Biology (BICoB 2011)*, March 2011.
- Program Committee, *10th IEEE International Workshop on High Performance Computational Biology (HiCOMB 2011)*, May 2011.
- Program Committee, *13th IEEE International Conference on High Performance Computing and Communications (HPCC-2011)*, September 2011.
- Program Committee, *6th International Conference on Bioinformatics and Computational Biology (BICoB)*, March 2014.

b **UNIVERSITY**

i **DEPARTMENT**

- Graduate Admissions Reviewer, 1996, 1999–2001, 2004–2006, 2009–2011.
- Judge, UMD High School Programming Contest, 1996–2001, 2003–2012, 2014.
- Director, UMD High School Programming Contest, 1997–2001, 2003–2010.
- Laboratory Committee, 1997.
- Teaching Committee, 1998.
- Graduate Student Review Committee, 2003–2009.
- Faculty Hiring Committee, 2003–2005.
- Introductory Teaching Committee, 2004–2007.
- Center for Bioinformatics and Computational Biology (CBCB) Faculty Hiring Committee, 2003–2004, 2005–2006.

- Information Technology Advisory Committee (ITAC), 2000–2001.
- University Senate, Fall 2006-Spring 2009, Spring 2010, Fall 2013-present.
- Campus Honor Council, Fall 2010–2012.

c **COMMUNITY**

- Board of Directors, Organization of Chinese Americans, Greater Washington DC Chapter (OCA-DC), 1997–2001.

I certify that this CV is accurate and complete — February 10, 2014

Chau-Wen Tseng