Donald Yeung
1323 A. V. Williams
College Park, MD 20742
voice: (301) 405-3649, fax: (301) 314-9281
http://www.ece.umd.edu/~yeung
yeung@eng.umd.edu

1 Personal Information

Birth Date: May 27, 1968.
Current Position: Associate Professor, Department of Electrical and Computer Engineering, University of Maryland at College Park.
Appointed to current position July 1, 2004.

a. Education

1998 Ph.D. Electrical Engineering and Computer Science, Massachusetts Institute of Technology. Dissertation: "Multigrain Shared Memory."

1993 S.M. Electrical Engineering and Computer Science, Massachusetts Institute of Technology. Master's Thesis: "An Evaluation of Multiprocessor Support for Fine-Grain Synchronization in Preconditioned Conjugate Gradient."

1990 B.S. Computer Systems Engineering, Stanford University.

b. Experience

2017-Present	Professor, Department of Electrical and Computer Engineering, University of Maryland at College Park.
2010-Present	Director of Computer Engineering Education, Department of Electrical and Computer Engineering, University of Maryland at College Park.
2004-2017	Associate Professor, Department of Electrical and Computer Engineering, University of Maryland at College Park.
2005	Visiting Research Scholar, Dynamic Systems Division, University of Southern California Information Sciences Institute, Arlington, VA.
1998-2004	Assistant Professor, Department of Electrical and Computer Engineering, University of Maryland at College Park.
1990-1997	Research Assistant, Department of Electrical Engineering and Computer Science, Massachussetts Institute of Technology.
1990	Teaching Assistant, Department of Electrical Engineering and Computer Science, Massachussetts Institute of Technology.
1988, 1990 Summers	Teaching Assistant, Summer Science Program, The Thacher School, Ojai, CA.
1987, 1988 Summers	Member of Technical Staff, Automated Manufacturing and Design Automation Department, TRW, Redondo Beach, CA.

2 Research and Scholarly Activities

a. Chapters in Books

A.1 David Kranz, Beng-Hong Lim, Donald Yeung, and Anant Agarwal. "Low-Cost Support for Fine-Grain Synchronization in Multiprocessors." *Multithreading: A Summary of the State of the Art.* Kluwer Academic Publishers. 1992.

A.2 Yan Solihin and Donald Yeung. "Data Cache Prefetching." *Speculative Execution in High Performance Computer Architectures.* CRC Press. 2005.

A.3 Janice McMahon, Steve Crago, and Donald Yeung. "Advanced Microprocessor Architectures." *High Performance Embedded Computing Handbook: A Systems Perspective.* CRC Press. 2008.

b. Articles in Refereed Journals

B.1 Anant Agarwal, John Kubiatowicz, David Kranz, Beng Hong Lim, Donald Yeung, Godfrey D'Souza, and Mike Parkin. "Sparcle: An Evolutionary Processor Design for Large-Scale Multiprocessors." *IEEE Micro.* pp. 48-61. June 1993. *Also appears in HOTCHIPS.* August 1992.

B.2 Anant Agarwal, Ricardo Bianchini, David Chaiken, Frederic T. Chong, Kirk L. Johnson, David Kranz, John D. Kubiatowicz, Beng-Hong Lim, Kenneth Mackenzie, and Donald Yeung. "The MIT Alewife Machine." *Proceedings of the IEEE.* Vol. 87, No. 3. pp. 430-444. March 1999.

B.3 Donald Yeung, John Kubiatowicz, and Anant Agarwal. "Multigrain Shared Memory." *ACM Transactions on Computer Systems.* Vol. 18, No. 2. pp. 154-196. May 2000.

B.4 Andras Moritz, Donald Yeung, and Anant Agarwal. "SimpleFit: A Framework for Analyzing Design Tradeoffs in Raw Architectures." *IEEE Transactions on Parallel and Distributed Systems.* Vol. 12, No. 6. pp. 730-742. June 2001.

B.5 Gautham K. Dorai, Donald Yeung, and Seungryul Choi. "Optimizing SMT Processors for High Single-Thread Performance." *Journal of Instruction-Level Parallelism.* Vol. 5. pp. 1-35. April 2003.

B.6 Seungryul Choi, Nicholas Kohout, Sumit Pamnani, Dongkeun Kim, and Donald Yeung. "A General Framework for Prefetch Scheduling in Linked Data Structures and its Application to Multi-Chain Prefetching." *ACM Transactions on Computer Systems.* Vol. 22, No. 2. pp. 214-280. May 2004.

B.7 Abdel-Hameed A. Badawy, Aneesh Aggarwal, Donald Yeung, and Chau-Wen Tseng. "The Efficacy of Software Prefetching and Locality Optimizations on Future Memory Systems." *Journal of Instruction-Level Parallelism.* Vol. 6. pp. 1-36. July 2004.

B.8 Dongkeun Kim and Donald Yeung. "A Study of Source-Level Compiler Algorithms for Automatic Construction of Pre-Execution Code." *ACM Transactions on Computer Systems*. Vol. 22, No. 3. pp. 326-379. August 2004.

B.9 Sumit Pamnani, Deepak Agarwal, Gang Qu, and Donald Yeung. "Low Power System Design with Performance Enhancement Techniques–General Approach and Case Study." *Journal of Circuits, Systems, and Computers*. Vol. 16, No. 5. pp. 745-767. October 2007.

B.10 Xuanhua Li and Donald Yeung. "Exploiting Application-Level Correctness for Low-Cost Fault Tolerance." *Journal of Instruction-Level Parallelism*. Vol. 10. pp. 1-28. September 2008.

B.11 Seungryul Choi and Donald Yeung. "Hill-Climbing SMT Processor Resource Distribution." *ACM Transactions on Computer Systems*. Vol. 27, No. 1. Article 1. pp. 1-47. February 2009.

B.12 Wanli Liu and Donald Yeung. "Enhancing LTP-Driven Cache Management Using Reuse Distance Information." *Journal of Instruction-Level Parallelism*. Vol. 11. pp. 1-24. April 2009.

B.13 Inseok Choi, Minshu Zhao, Xu Yang, and Donald Yeung. "Experience with Improving Distributed Shared Cache Performance on Tilera's Tile Processor." *IEEE Computer Architecture Letters*. Vol. 10, Issue 1. pp. 45-48. July 2011.

B.14 Meng-Ju Wu and Donald Yeung. "Efficient Reuse Distance Analysis of Multicore Scaling for Loop-based Parallel Programs." *ACM Transactions on Computer Systems*. Vol. 31, No. 1. Article 1. pp. 1-37. February 2013.

B.15 Caleb Serafy, Avram Bar-Cohen, Ankur Srivastava, and Donald Yeung. "Unlocking the True Potential of 3D CPUs with Micro-Fluidic Cooling." *IEEE Transactions on Very Large Scale Integration Systems*. Vol. 24, No. 4. pp. 1515-1523. April 2016.

B.16 Mike Badamo, Jeff Casarona, Minshu Zhao, and Donald Yeung. "Identifying Power Efficient Multicore Cache Hierarchies via Reuse Distance Analysis." *ACM Transactions on Computer Systems*. Vol. 34, No. 1. Article 3. pp. 1-30. April 2016.

B.17 I. Stephen Choi and Donald Yeung. "Multi-Cache Resizing via Greedy Coordinate Descent." *Journal of Supercomputing*. Vol. 73, No. 6. pp. 2402-2429. June 2017.

B.18 Abdel-Hameed A. Badawy and Donald Yeung. "Guiding Locality Optimizations for Graph Computations via Reuse Distance Analysis." To appear in *IEEE Computer Architecture Letters*.

B.19 Minshu Zhao and Donald Yeung. "Using Multicore Reuse Distance to Study Coherence Directories." *ACM Transactions on Computer Systems*. Vol. 35, No. 2. Article 4. July 2017.

c. Articles in Refereed Symposia, Conferences, and Workshops

i. Highly Competitive Conferences. In the Computer Systems field, publications in the top journals take a year or more from submission to publication. Given the fast pace of this field, the long publishing delay for journals reduces the impact of research reported in journal articles. Consequently, high-quality conferences, and *not* journals, are the primary avenue for disseminating important ideas in Computer Systems. To ensure the quality of articles appearing in highly competitive conferences, full manuscripts (6000-9000 word papers, not extended abstracts) must be submitted. Furthermore, the acceptance rates are low, around 20% for the most competitive conferences.

C.1 Donald Yeung and Anant Agarwal. "Experience with Fine-Grain Synchronization in MIMD Machines for Preconditioned Conjugate Gradient." In *Proceedings of the Fourth ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (PPoPP-IV).* San Diego, California. May 1993.
(26 papers accepted out of 96 papers submitted–27% acceptance rate)

C.2 Anant Agarwal, Ricardo Bianchini, David Chaiken, Kirk Johnson, David Kranz, John Kubiatowicz, Beng-Hong Lim, Kenneth Mackenzie, and Donald Yeung. "The MIT Alewife Machine: Architecture and Performance." In *Proceedings of the 22nd International Symposium on Computer Architecture (ISCA-XXII).* Santa Margherita, Italy. June 1995.
(37 papers accepted out of 180 papers submitted–21% acceptance rate)

C.3 Donald Yeung, John Kubiatowicz, and Anant Agarwal. "MGS: A Multigrain Shared Memory System." In *Proceedings of the 23rd ACM SIGPLAN International Symposium on Computer Architecture (ISCA-XXIII).* Philadelphia, Pennsylvania. May 1996.
(28 papers accepted out of 112 papers submitted–25% acceptance rate)

C.4 Andras Moritz, Donald Yeung, and Anant Agarwal. "Exploring Cost-Performance Optimal Designs of Raw Microprocessors." In *Proceedings of the 6th Annual IEEE Symposium on Field-Programmable Custom Computing Machines (FCCM-VI).* Napa, California. April 1998.
(25 papers accepted out of 87 papers submitted–29% acceptance rate)

C.5 Donald Yeung. "The Scalability of Multigrain Systems." In *Proceedings of the 13th Annual International Conference on Supercomputing (ICS-XIII).* Rhodes, Greece. June 1999.
(57 papers accepted out of 180 papers submitted–32% acceptance rate)

C.6 Nicholas Kohout, Seungryul Choi, and Donald Yeung. "Multi-Chain Prefetching: Exploiting Memory Parallelism in Pointer-Chasing Codes." *Solving the Memory Wall Problem Workshop, held in conjunction with ISCA-XXVII.* Vancouver, Canada. June 2000.
(13 papers accepted out of 40 papers submitted–33% acceptance rate)

C.7 Abdel-Hameed A. Badawy, Aneesh Aggarwal, Donald Yeung, and Chau-Wen Tseng. "Evaluating the Impact of Memory System Performance on Software Prefetching and Locality Optimizations." In *Proceedings of the 15th Annual International Conference on Supercomputing (ICS-XV).* Sorrento, Italy. June 2001.
(45 papers accepted out of 133 papers submitted–34% acceptance rate)

C.8 Nicholas Kohout, Seungryul Choi, Dongkeun Kim, and Donald Yeung. "Multi-Chain Prefetching: Effective Exploitation of Inter-Chain Memory Parallelism for Pointer-Chasing

Codes." In *Proceedings of the 10th Annual International Conference on Parallel Architectures and Compilation Techniques (PACT-X)*. Barcelona, Spain. September 2001.
(26 papers accepted out of 126 papers submitted–21% acceptance rate)

C.9 Gautham K. Dorai and Donald Yeung. "Transparent Threads: Resource Allocation in SMT Processors for High Single-Thread Performance." In *Proceedings of the 11th Annual International Conference on Parallel Architectures and Compilation Techniques (PACT-XI)*. Charlottesville, VA. September 2002.
(25 papers accepted out of 119 papers submitted–21% acceptance rate)

C.10 Dongkeun Kim and Donald Yeung. "Design and Evaluation of Compiler Algorithms for Pre-Execution." In *Proceedings of the 10th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS-X)*. San Jose, CA. October 2002.
(24 papers accepted out of 130 papers submitted–18% acceptance rate)

C.11 Dongkeun Kim, Steve Shih-wei Liao, Perry Wang, Juan del Cuvillo, Xinmin Tian, Xiang Zou, Hong Wang, Donald Yeung, Milind Girkar, and John Shen. "Physical Experimentation with Prefetching Helper Threads on Intel's Hyper-Threaded Processors." In *Proceedings of the 2004 International Symposium on Code Generation and Optimization with Special Emphasis on Feedback-Directed and Runtime Optimization (CGO-II)*. San Jose, CA. March 2004.
(25 papers accepted out of 79 papers submitted–32% acceptance rate)

C.12 Kursad Albayraktaroglu, Aamer Jaleel, Xue Wu, Bruce Jacob, Manoj Franklin, Chau-Wen Tseng, and Donald Yeung. "BioBench: A Benchmark Suite of Bioinformatics Applications." In *Proceedings of the IEEE International Symposium on Performance Analysis of Systems and Software (ISPASS-V)*. Austin, TX. March 2005.
(27 papers accepted out of 92 papers submitted–29% acceptance rate)

C.13 Seungryul Choi and Donald Yeung. "Learning-Based SMT Processor Resource Distribution via Hill-Climbing." In *Proceedings of the 33rd International Symposium on Computer Architecture (ISCA-XXXIII)*. Boston, MA. June 2006.
(31 papers accepted out of 229 papers submitted–14% acceptance rate)

C.14 Xuanhua Li and Donald Yeung. "Application-Level Correctness and its Impact on Fault Tolerance." In *Proceedings of the 13th International Symposium on High Performance Computer Architecture (HPCA-XIII)*. Phoenix, AZ. February 2007.
(28 papers accepted out of 174 papers submitted–16% acceptance rate)

C.15 Wanli Liu and Donald Yeung. "Using Aggressor Thread Information to Improve Shared Cache Management for CMPs." In *Proceedings of the 18th International Conference on Parallel Architectures and Compilation Techniques (PACT-XVIII)*. Raleigh, NC. September 2009.
(35 papers accepted out of 188 papers submitted–19% acceptance rate)

C.16 Meng-Ju Wu and Donald Yeung. "Coherent Profiles: Enabling Efficient Reuse Distance Analysis of Multicore Scaling for Loop-based Parallel Programs." In *Proceedings of the 20th International Conference on Parallel Architectures and Compilation Techniques (PACT-XX)*. Galveston Island, TX. October 2011.
(36 papers accepted out of 221 papers submitted–16% acceptance rate)

C.17 Meng-Ju Wu, Minshu Zhao, and Donald Yeung. "Studying Multicore Processor Scaling via Reuse Distance Analysis." In *Proceedings of the 40th International Symposium on Computer Architecture (ISCA-XL).* Tel-Aviv, Israel. June 2013.
(56 papers accepted out of 288 papers submitted–19% acceptance rate)

C.18 Caleb Serafy, Ankur Srivastava, and Donald Yeung. "Unlocking the True Potential of 3D CPUs with Micro-Fluidic Cooling." In *Proceedings of the International Symposium on Low Power Electronics and Design (ISLPED'14).* La Jolla, California. August 2014.
(63 papers accepted out of 184 papers submitted–34% acceptance rate)

C.19 Minshu Zhao and Donald Yeung. "Studying the Impact of Multicore Processor Scaling on Directory Techniques via Reuse Distance Analysis." In *Proceedings of the 21st International Symposium on High Performance Computer Architecture (HPCA-XXI).* San Francisco Bay Area, CA. February 2015.
(51 papers accepted out of 226 papers submitted–23% acceptance rate)

ii. Other Refereed Symposia, Conferences, and Workshops.

C.20 John Kubiatowicz, David Chaiken, Anant Agarwal, Arthur Altman, Jonathan Babb, David Kranz, Beng-Hong Lim, Ken Mackenzie, John Piscitello, and Donald Yeung. "The Alewife CMMU: Addressing the Multiprocessor Communications Gap." *Hot Chips: A Symposium on High Performance Chips.* Stanford, CA. August 1994.

C.21 Donald Yeung, Nicholas Kohout, Sujata Ramasubramanian, Ilya Khazanov, and Rishi Kurichh. "Vortex: Irregular Data Stream Support for Data-Intensive Applications." *Eighth Scalable Shared Memory Multiprocessors Workshop, held in conjunction with ISCA-XXVI.* Atlanta, GA. April 1999.

C.22 Deepak Agarwal, Wanli Liu, and Donald Yeung. "Exploiting Application-Level Information to Reduce Memory Bandwidth Consumption." In *Proceedings of the 4th Workshop on Complexity-Effective Design, held in conjunction with ISCA-XXX.* San Diego, CA. June 2003.

C.23 Deepak Agarwal, Sumit Pamnani, Gang Qu, and Donald Yeung. "Transferring Performance Gain from Prefetching to Energy Reduction." In *Proceedings of the International Symposium on Circuits and Systems.* Vancouver, Canada. May 2004.

C.24 Xuanhua Li and Donald Yeung. "Exploiting Soft Computing for Increased Fault Tolerance." In *Proceedings of the 2006 Workshop on Architectural Support for Gigascale Computing.* Boston, MA. June 2006.

C.25 Xuanhua Li and Donald Yeung. "Exploiting Value Prediction for Fault Tolerance." In *Proceedings of the 3rd Workshop on Dependable Architectures.* Lake Como, Italy. November 2008.

C.26 Inseok Choi, Minshu Zhao, Xu Yang, and Donald Yeung. "Early Experience with Profiling and Optimizing Distributed Shared Cache Performance on Tilera's Tile Processor." In *Proceedings of the 6th International Workshop on Unique Chips and Systems.* Atlanta, GA. December 2010. **One of 2 best papers chosen out of 12 papers appearing in the workshop.**

C.27 Eric Lau, Jason Miller, Inseok Choi, Donald Yeung, Saman Amarasinghe, and Anant Agarwal. "Multicore Performance Optimization using Positive Energy Partnerships." In *Proceedings of the 3rd USENIX Workshop on Hot Topics in Parallelism (HotPar '11).* Berkeley, CA. May 2011.

C.28 Meng-Ju Wu and Donald Yeung. "Identifying Optimal Multicore Cache Hierarchies for Loop-based Parallel Programs via Reuse Distance Analysis." In *Proceedings of the ACM SIGPLAN Workshop on Memory Systems Performance and Correctness (MSPC-2012).* Beijing, China. June 2012.

C.29 Caleb Serafy, Bing Shi, Ankur Srivastava, and Donald Yeung. "High Performance 3D Stacked DRAM Processor Architectures with Micro-Fluidic Cooling." In *Proceedings of the IEEE 3D System Integration Conference 2013 (Poster Session).* San Francisco, California. October 2013.

C.30 Caleb Serafy, Ankur Srivastava, Bing Shi, and Donald Yeung. "Continued Frequency Scaling in 3D ICs through Micro-fluidic Cooling." In *Proceedings of the Intersociety Conference on Thermal and Thermomechanical Phenomena in Electronic Systems (iTHERM '14).* Orlando, Florida. May 2014.

C.31 Caleb Serafy, Bing Shi, Ankur Srivastava, and Donald Yeung. "Electro-Thermo Co-Design for Micro-Fluidically Cooled 3D ICs." In *Proceedings of the 51st Design Automation Conference, Work-In-Progress Session.* San Francisco, California. June 2014.

C.32 Caleb Serafy, Ankur Srivastava, Avram Bar-Cohen and Donald Yeung. "Design Space Exploration of 3D CPUs and Micro-Fluidic Heatsinks with Thermo-Electrical-Physical Co-Optimization." In *Proceedings of the ASME 2015 International Technical Conference and Exhibition on Packaging and Integration of Electronic and Photonic Microsystems, and 13th International Conference on Nanochannels, Microchannels, and Minichannels.* San Francisco, California. July 2015.

C.33 Stephen P. Crago and Donald Yeung. "Reducing Data Movement with Approximate Computing Techniques." In *Proceedings of the IEEE International Conference on Rebooting Computing.* San Diego, CA. October 2016.

C.34 Michael Zuzak and Donald Yeung. "Exploiting Multi-Loop Parallelism on Heterogeneous Microprocessors." In *Proceedings of the 10th International Workshop on Programmability and Architectures for Heterogeneous Multicores (MULTIPROG-2017) held in conjunction with HiPEAC-12.* Stockholm, Sweden. January 2017. **Best paper award**.

iii. Invited Conferences and Workshops

C.35 Stephen Crago, Janice Onanian McMahon, Chris Archer, Krste Asanovic, Richard Chaung, Keith Goolsbey, Mary Hall, Christos Kozyrakis, Kunle Olukotun, Una-May O'Reilly, Rick Pancoast, Viktor Prasanna, Rodric Rabbah, Steve Ward, and Donald Yeung. "CEARCH: Cognition-Enabled Architecture." In *Proceedings of the 10th Annual High Performance Embedded Computing Workshop.* Lexington, MA. September 2006.

C.36 Meng-Ju Wu, Minshu Zhao, Mike Badamo, Jeff Casarona, and Donald Yeung. "Characterizing Embedded Applications via Multicore Reuse Distance Analysis." In *Workshop on Suite of Embedded Applications and Kernels (Poster Session), held in conjunction with DAC-LI.* San Francisco, CA. June 2014.

e. Technical Reports

E.1 Donald Yeung and Anant Agarwal. "Experience with Fine-Grain Synchronization in MIMD Machines for Preconditioned Conjugate Gradient." *MIT/LCS Technical Memo, MIT-LCS-TM-479.* October 1992.

E.2 Donald Yeung. "An Evaluation of Multiprocessor Support for Fine-Grain Synchronization in Preconditioned Conjugate Gradient." Master's Thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology. *MIT/LCS Technical Report, MIT-LCS-TR-565.* February 1993.

E.3 Donald Yeung, William Dally, and Anant Agarwal. "How to Choose the Grain Size of a Parallel Computer." *MIT/LCS Technical Report, MIT-LCS-TR-739.* February 1994.

E.4 Donald Yeung. "Multigrain Shared Memory." Ph.D. Thesis, Department of Electrical Engineering and Computer Science, Massachusetts Institute of Technology. *MIT/LCS Technical Report, MIT-LCS-TR-743.* January 1998.

E.5 Nicholas Kohout, Seungryul Choi, and Donald Yeung. "Multi-Chain Prefetching: Exploiting Memory Parallelism in Pointer-Chasing Codes." *University of Maryland Systems and Computer Architecture Group Technical Report, UMD-SCA-TR-2000-01.* June 2000.

E.6 Aneesh Aggarwal, Abdel-Hameed A. Badawy, Donald Yeung, and Chau-Wen Tseng. "Evaluating the Impact of Memory System Performance on Software Prefetching and Locality Optimizations." *University of Maryland Institute for Advanced Computer Studies Technical Report, UMIACS-TR-2000-57.* July 2000.

E.7 Dongkeun Kim and Donald Yeung. "Using Program Slicing to Drive Pre-Execution on Simultaneous Multithreading Processors." *University of Maryland Institute for Advanced Computer Studies Technical Report, UMIACS-TR-2001-49.* June 2001.

E.8 Deepak Agarwal and Donald Yeung. "Exploiting Application-Level Information to Reduce Memory Bandwidth Consumption." *University of Maryland Institute for Advanced Computer Studies Technical Report, UMIACS-TR-2002-64.* July 2002.

E.9 Gautham K. Dorai, Donald Yeung, and Seungryul Choi. "Optimizing SMT Processors for High Single-Thread Performance." *University of Maryland Institute for Advanced Computer Studies Technical Report, UMIACS-TR-2003-07.* January 2003.

E.10 Seungryul Choi and Donald Yeung. "Hill-Climbing SMT Processor Resource Scheduler." *University of Maryland Institute for Advanced Computer Studies Technical Report, UMIACS-TR-2005-30.* May 2005.

E.11 Xuanhua Li and Donald Yeung. "Application-Level Correctness and its Impact on Fault Tolerance." *University of Maryland Institute for Advanced Computer Studies Technical Report, UMIACS-TR-2006-36.* August 2006.

E.12 Meng-Ju Wu and Donald Yeung. "Parallelization of the SSCA #3 Benchmark on the RAW Processor." *University of Maryland Institute for Advanced Computer Studies Technical Report, UMIACS-TR-2006-42.* August 2006.

E.13 Wanli Liu and Donald Yeung. "Enhancing LTP-Driven Cache Management Using Reuse Distance Information." *University of Maryland Institute for Advanced Computer Studies Technical Report, UMIACS-TR-2007-33.* June 2007.

E.14 Wanli Liu and Donald Yeung. "Compatible Working Sets: the Case for Flexible Management of Shared Caches in CMPs." *University of Maryland Institute for Advanced Computer Studies Technical Report, UMIACS-TR-2008-13.* July 2008.

E.15 Meng-Ju Wu and Donald Yeung. "Scaling Single-Program Performance on Large-Scale Chip Multiprocessors." *University of Maryland Institute for Advanced Computer Studies Technical Report, UMIACS-TR-2009-16.* November 2009.

E.16 Meng-Ju Wu and Donald Yeung. "Memory Performance Analysis for Parallel Programs Using Concurrent Reuse Distance." *University of Maryland Institute for Advanced Computer Studies Technical Report, UMIACS-TR-2010-10.* October 2010.

E.17 Meng-Ju Wu and Donald Yeung. "Understanding Multicore Cache Behavior of Loop-based Parallel Programs via Reuse Distance Analysis." *University of Maryland Institute for Advanced Computer Studies Technical Report, UMIACS-TR-2012-01.* January 2012.

E.18 Inseok Choi and Donald Yeung. "Multi-Level Cache Resizing." *University of Maryland Institute for Advanced Computer Studies Technical Report, UMIACS-TR-2012-11.* November 2012.

E.19 Inseok Choi and Donald Yeung. "Symbiotic Cache Resizing for CMPs with Shared LLC." *University of Maryland Institute for Advanced Computer Studies Technical Report, UMIACS-TR-2013-02.* Septebmer 2013.

E.20 Minshu Zhao and Donald Yeung. "Studying Directory Access Patterns via Reuse Distance Analysis and Evaluating Their Impact on Multi-Level Directory Caches." *University of Maryland Institute for Advanced Computer Studies Technical Report, UMIACS-TR-2014-01.* January 2014.

E.21 Michael Zuzak and Donald Yeung. "Exploiting Multi-Loop Parallelism on Heterogeneous Microprocessors." *University of Maryland Institute for Advanced Computer Studies Technical Report, UMIACS-TR-2016-01.* November 2014.

f. Presentations

i. Invited Talks

F.1 "MGS: A Multigrain Shared Memory System." *ARPA site visit, MIT Laboratory for Computer Science.* Cambridge, MA. September 1995.

F.2 "The Alewife Machine: A Scalable Distributed Shared Memory Multiprocessor." *Parallel Computing Seminar, University of Massachusetts at Boston.* Boston, MA. May 1996.

F.3 "MGS: A Multigrain Shared Memory System." *Institute for Advanced Computer Studies Research Meeting, University of Maryland.* College Park, MD. May 1998.

F.4 "MGS: A Multigrain Shared Memory System." *Computer Engineering Seminar Series, University of Maryland.* College Park, MD. November 1998.

F.5 "Compiler and Architecture Support for Sub-Ordinate Multithreading." *Advanced Processor Architecture Group, Sun Microsystems.* Sunnyvale, CA. October 2002.

F.6 "Compiler and Architecture Support for Sub-Ordinate Multithreading." *Itanium Group, Intel Corporation.* Santa Clara, CA. October 2002.

F.7 "Architecture and Compiler Support for Speculative Precomputation." Half-day tutorial to be presented with Prof. Dean Tullsen (UC San Diego) and Dr. Steve Shih-wei Liao (Intel Corp) at the *12th Annual International Conference on Parallel Architectures and Compilation Techniques.* New Orleans, LA. September 2003.

F.8 "Improving Memory and Parallel Performance via Helper Threads." *Computer Architecture Group, IBM T. J. Watson.* Yorktown Heights, NY. October 2004.

F.9 "Accessing PMCs on Maestro," presented at the Workshop on Multicore Processors for Space Opportunities and Challenges held in conjunction with the IEEE International Conference on Space Mission Challenges for Information Technology. Pasadena, CA. July 2009.

F.10 "OpenJPEG2K Performance on Maestro," presented at the Workshop on Multicore Processors for Space Opportunities and Challenges held in conjunction with the IEEE International Conference on Space Mission Challenges for Information Technology. Pasadena, CA. July 2009.

F.11 "The Angstrom Project: Fundamental Technologies for Manycore Computing," *Hughes Network Systems Advanced Development Group Seminar.* Gaithersburg, MD. September 2011.

F.12 "Studying Multicore Memory Behavior using Reuse Distance Analysis," *Computer Architecture Group, IBM T. J. Watson.* Yorktown Heights, NY. October 2015.

ii. Contributed Talks

F.13 "Experience with Fine-Grain Synchronization in MIMD Machines for Preconditioned Conjugate Gradient," presented at the *4th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming.* San Diego, California. May 1993.

F.14 "MGS: A Multigrain Shared Memory System," presented at the *23rd ACM SIGPLAN International Symposium on Computer Architecture.* Philadelphia, PA. May 1996.

F.15 "Vortex: Irregular Data Stream Support for Data-Intensive Applications," presented at the *Eighth Scalable Shared Memory Multiprocessors Workshop.* Atlanta, GA. April 1999.

F.16 "The Scalability of Multigrain Systems," presented at the *13th Annual International Conference on Supercomputing.* Rhodes, Greece. June 1999.

F.17* "Multi-Chain Prefetching: Exploiting Memory Parallelism in Pointer-Chasing Codes," presented at *Solving the Memory Wall Problem Workshop.* Vancouver, Canada. June 2000.

F.18* "Evaluating the Impact of Memory System Performance on Software Prefetching and Locality Optimizations," presented at the *15th Annual International Conference on Supercomputing.* Sorrento, Italy. June 2001.

F.19* "Multi-Chain Prefetching: Effective Exploitation of Inter-Chain Memory Parallelism for Pointer-Chasing Codes," presented at the *10th Annual International Conference on Parallel Architectures and Compilation Techniques.* Barcelona, Spain. September 2001.

F.20* "Transparent Threads: Resource Allocation in SMT Processors for High Single-Thread Performance," presented at the *11th Annual International Conference on Parallel Architectures and Compilation Techniques.* Charlottesville, VA. September 2002.

F.21* "Design and Evaluation of Compiler Algorithms for Pre-Execution," presented at the *10th International Conference on Architectural Support for Programming Languages and Operating Systems.* San Jose, CA. October 2002.

F.22* "Exploiting Application-Level Information to Reduce Memory Bandwidth Consumption," presented at the *4th Workshop on Complexity-Effective Design.* San Diego, CA. June 2003.

F.23 "Detailed CEARCH Kernel Analysis," presented at the DARPA Architectures for Cognitive Information Processing program review of the Information Sciences Institute team. Arlington, VA. September 2005.

F.24 "Soft Computing and Adaptive Machine Grain Size," presented at the DARPA Architectures for Cognitive Information Processing program review of the Information Sciences Institute team. Arlington, VA. September 2005.

F.25 "CEARCH Kernel Analysis and its Architectural Implications," presented at the DARPA Architectures for Cognitive Information Processing program review of the Information Sciences Institute team. Arlington, VA. January 2006.

F.26* "Exploiting Soft Computing for Increased Fault Tolerance," presented at the *2006 Workshop on Architectural Support for Gigascale Computing.* Boston, MA. June 2006.

F.27* "Learning-Based SMT Processor Resource Distribution via Hill-Climbing," presented at the *33rd International Symposium on Computer Architecture.* Boston, MA. June 2006.

F.28 "Soft Computing: Hardware, Software, and Programming Model Implications," presented at the DARPA Architectures for Cognitive Information Processing program review of the Information Sciences Institute team. Hartford, CT. July 2006.

F.29* "Application-Level Correctness and its Impact on Fault Tolerance," presented at the *13th International Symposium on High Performance Computer Architecture.* Phoenix, AZ. February 2007.

F.30 "VSIPL Implementation for the Tilera Multicore Processor," presented at the NRO Opera program review of the Information Sciences Institute team. Arlington, VA. August 2008.

F.31 "JPEG2000 Application Study on the Tilera Multicore Processor," presented at the NRO Opera program review of the Information Sciences Institute team. Arlington, VA. August 2008.

F.32 "Application-Specific Architectures for Bioinformatics Algorithms" presented at the DARPA Algorithm Classes for Architecture Research program review of the Information Sciences Institute team. Arlington, VA. October 2008.

F.33 "Exploiting Value Prediction for Fault Tolerance," presented at the *3rd Workshop on Dependable Architectures.* Lake Como, Italy. November 2008.

F.34 "VSIPL / JPEG2K Release," presented at the NRO Opera program review of the Information Sciences Institute team. Arlington, VA. April 2009.

F.35 "Parallel Performance Analysis Tools for MAESTRO," presented at the NRO Opera program review of the Information Sciences Institute team. Arlington, VA. April 2009.

F.36 "Analyzing Memory Bottlenecks on Maestro," presented at the NRO Opera program review of the Information Sciences Institute team. Arlington, VA. July 2009.

F.37* "Using Aggressor Thread Information to Improve Shared Cache Management for CMPs," presented at the *18th International Conference on Parallel Architectures and Compilation Techniques.* Raleigh, NC. September 2009.

F.38 "PAPI and Perfmon2 Implementation on Maestro," presented at the NRO Opera program review of the Information Sciences Institute team. Arlington, VA. December 2009.

F.39 "mProf: A Tool for Analyzing Memory Bottlenecks on Maestro," presented at the NRO Opera program review of the Information Sciences Institute team. Arlington, VA. December 2009.

F.40 "Papi on MDE 2.1 and Locality Optimization Using mProf," presented at the NRO Opera program review of the Information Sciences Institute team. Arlington, VA. April 2010.

F.41 "Papi and mProf-p Software Releases," presented at the NRO Opera program review of the Information Sciences Institute team. Arlington, VA. July 2010.

F.42 "Helper Threads and Soft Computing for Angstrom," presented at the DARPA UHPC program kick-off meeting. Cambridge, MA. July 2010.

F.43* "Early Experience with Profiling and Optimizing Distributed Shared Cache Performance on Tilera's Tile Processor," presented at the *6th International Workshop on Unique Chips and Systems.* Atlanta, GA. December 2010.

F.44 "Streaming Graphs and Helper Threads for Angstrom," presented at the DARPA UHPC program annual review. Cambridge, MA. September 2011.

F.45* "Coherent Profiles: Enabling Efficient Reuse Distance Analysis of Multicore Scaling for Loop-based Parallel Programs," presented at the *20th International Conference on Parallel Architectures and Compilation Techniques.* Galveston Island, TX. October 2011.

F.46* "Identifying Optimal Multicore Cache Hierarchies for Loop-based Parallel Programs via Reuse Distance Analysis," presented at the *ACM SIGPLAN Workshop on Memory Systems Performance and Correctness.* Beijing, China. June 2012.

F.47 "Streaming Graph Challenge Problem," presented at the DARPA UHPC program annual review. Cambridge, MA. August 2012.

F.48 "UHPC Challenge Apps," presented at the DARPA UHPC program quarterly review. Cambridge, MA. November 2012.

F.49 "Multicore Locality Analysis and Optimization," presented at the DARPA PERFECT program brainstorming meeting. Cambridge, MA. December 2012.

F.50 "Multicore Locality Analysis and Optimization," presented at the DARPA PERFECT bi-weekly program review. February 2013.

F.51* "Studying Multicore Processor Scalability via Reuse Distance Analysis," presented at the *40th International Symposium on Computer Architecture.* Tel-Aviv, Israel. June 2013.

F.52 "UHPC Challenge Problems," presented at the DARPA UHPC program annual review. Cambridge, MA. July 2013.

F.53 "Parallel Program Locality Profiles," presented at the DARPA PERFECT bi-weekly program review. July 2013.

F.54 "Parallel Program Locality Profiles," presented at the DARPA PERFECT bi-weekly program review. December 2013.

F.55* "Studying the Impact of Multicore Processor Scaling on Directory Techniques via Reuse Distance Analysis," presented at the *21st International Symposium on High Performance Computer Architecture.* San Francisco Bay Area, CA. February 2015.

F.56 "Runtime and Memory Systems Research," presented at the *Advanced Computing Systems Group, Laboratory for Physical Sciences, University of Maryland at College Park.* Baltimore, MD. February 2015.

F.57 "Approximate Computing," presented at the IARPA Visit Day. College Park, MD. February 2016.

F.58* "Exploiting Multi-Loop Parallelism on Heterogeneous Microprocessors," presented at the *10th International Workshop on Programmability and Architectures for Heterogeneous Multicores.* Stockholm, Sweden. January 2017.

F.59 "RRAM Memory Systems and the New CPU Architectures They Enable," presented at the Army Research Laboratory. Aberdeen, MD. March 2017.

F.60 "RRAM: Enabling CPUs with On-chip Memory Systems," presented at the Laboratory for Physical Sciences. Baltimore, MD. June 2017.

*These conference talks were given by the student lead author listed in Section C. for the corresponding conference paper. However, in all cases, I was closely involved in creating the slides, and preparing the student through numerous practice talks.

g. Contracts and Grants

G.1 National Science Foundation, Computer Systems Architecture Program. "Uncovering and Exploiting Memory Parallelism in Pointer-Chasing Applications," $320,000, Principal Investigator. Co-PI: Chau-Wen Tseng. July 2000 – June 2003.

G.2 National Science Foundation, Computer Systems Architecture Program. "CAREER: Closing the Memory Gap for Unstructured Applications," $250,000, Principal Investigator. February 2001 – January 2006.

G.3 National Science Foundation, Information Technology Research Program. "PRAM-On-Chip." Total: $750,000. My share: $0, Senior Personnel. PI: Uzi Vishkin. 2003 – 2008.

G.4 Defense Advanced Research Projects Agency, Architectures for Cognitive Information Processing (ACIP) program. "Algorithmic and Architectural Techniques for Cognitive Processing." Total: $6,501,751. My share: $196,634 (also see contract G.8), Principal Investigator for University of Maryland. Prime: University of Southern California Information Sciences Institute (also with Massachusetts Institute of Technology and Stanford University). August 2005 – July 2006.

G.5 Defense Advanced Research Projects Agency. "Impact of Radiation-Hard Techniques on Performance in the MIT RAW Microprocessor." My share: $43,587, Principal Investigator for University of Maryland. Prime: University of Southern California Information Sciences Institute. August 2005 – December 2005.

G.6 IBM Shared University Research Award. "Evaluating Helper Threads on the IBM POWER 5 Architecture." $14,129 (equipment), Principal Investigator. November 2005 – October 2006.

G.7 Air Force Research Laboratory. "Benchmark Analysis for the X-Map Infrastructure." Total: $1,475,899. My share: $62,760 (also see contract G.9), Principal Investigator for University of Maryland. Prime: University of Southern California Information Sciences Institute. February 2006 – June 2006.

G.8 Defense Advanced Research Projects Agency, Architectures for Cognitive Information Processing (ACIP) program. Extension of contract for "Algorithmic and Architectural Techniques for Cognitive Processing." $30,000, Principal Investigator for University of Maryland. Prime: University of Southern California Information Sciences Institute. July 2006 – December 2006.

G.9 Air Force Research Laboratory. Extension of contract for "Benchmark Analysis for the X-Map Infrastructure." $64,470, Principal Investigator for University of Maryland. Prime: University of Southern California Information Sciences Institute. September 2006 – March 2007.

G.10 U.S. Government, OPERA Software Architecture (OSA) program. "Software and Performance Analysis for the Tilera Multicore Processor." Total: $8,900,000. My share: $160,305 (also see contracts G.11, G.13, and G.14), Principal Investigator for University of Maryland. Prime: University of Southern California Information Sciences Institute. April 2008 – November 2008.

G.11 U.S. Government, OPERA Software Architecture (OSA) program. "Software and Performance Analysis of Image Compression Algorithms on the Tilera Multicore Processor." My share: $28,000, Principal Investigator for University of Maryland. Prime: University of Southern California Information Sciences Institute. April 2008 – November 2008.

G.12 Defense Advanced Research Projects Agency. "Identifying Key Algorithm Classes in the CEARCH and BioBench Benchmark Suites." Total: $725,000. My share: $68,221, Principal Investigator for University of Maryland. Prime: University of Southern California Information Sciences Institute. May 2008 – December 2008.

G.13 U.S. Government, OPERA Software Architecture (OSA) program. "Parallel Performance Analysis on the Tilera Multicore Processor." My share: $189,749, Principal Investigator for University of Maryland. Prime: University of Southern California Information Sciences Institute (also with Reservoir Labs). January 2009 – December 2009.

G.14 U.S. Government, OPERA Software Architecture (OSA) program. Extension of contract for "Parallel Performance Analysis on the Tilera Multicore Processor." My share: $143,326, Principal Investigator for University of Maryland. Prime: University of Southern California Information Sciences Institute. January 2010 – September 2010.

G.15 Defense Advanced Research Projects Agency, Ubiquitous High-Performance Computing (UHPC) program. "The Angstrom Project: Universal Technologies for Exascale Computing." Total: $14,000,000. My share: $695,764, Principal Investigator for University of Maryland. Prime: Massachusetts Institute of Technology (also with Freescale Semiconductor). July 2010 – June 2013.

G.16 Defense Advanced Research Projects Agency, Power Efficiency Revolution for Embedded Computing Technologies (PERFECT) program. "Carbon: Organic Computing for Embedded Applications." Total: $2,100,000. My share: $250,000, Principal Investigator for University of Maryland. Prime: Massachusetts Institute of Technology (also with Princeton and University of Chicago). November 2013 – May 2014.

G.17 National Science Foundation, Software and Hardware Foundations Program. "Developing and Applying Reuse Distance Analysis Techniques for Large-Scale Multicore Processors," Total: $499,998. My share: $300,000, Principal Investigator. Co-PI: Ankur Srivastava. July 2011 – June 2016.

G.18 Office of Naval Research. "A STEM Workforce Development and Recruitment System," Total: $850,000. My share: $20,790, Principal Investigator for Dept. of Electrical Engineering at the University of Maryland. Prime: The Patuxent Partnership (also with the Naval Air Warefare Center-Aircraft Division, St. Mary's County Public Schools, St. Mary's College of Maryland, and the Dept. of Mechanical Engineering at the University of Maryland). July 2015 – June 2017.

G.19 National Science Foundation, Software and Hardware Foundations Program. "Parallelization and Memory System Techniques for Heterogeneous Microprocessors," Total: $400,000. Principal Investigator. August 2016 – July 2019.

University of Maryland Internal:

G.20 Minta Martin Aeronautical Research Fund. "Proposal for Minta Martin Funding," $35,000, Principal Investigator. January 1998.

G.21 Graduate Research Board Semester Award. "Architectural Support for Chip Multiprocessors," $9,250, Principal Investigator. August 2004.

h. Awards

- George Corcoran Memorial Award, May 2000.

  "Presented to Professor Donald Yeung for significant contributions to Electrical and Computer Engineering Education, in recognition of teaching and education leadership at the College Park campus, effective contributions at the national level, and creative and other scholarly activities related to Electrical and Computer Engineering education."

- National Science Foundation Career Award, February 2001.

  "The Faculty Early Career Development (CAREER) Program ... offers the National Science Foundation's most prestigious awards for new faculty members. The CAREER program recognizes and supports the early career-development activities of those teacher-scholars who are most likely to become the academic leaders of the 21st century."

- IBM Shared University Research Award, November 2005.

i. Selected Examples of Research Impact

My research can be grouped into six major areas: multicore reuse distance, approximate or soft computing, dynamic resource allocation, helper threads, pointer prefetching, and multiprocessor clusters. The following summarizes the most significant aspects of this research and their impact.

Multicore Reuse Distance. The performance and power consumption of multicore processors is determined in large part by how effectively applications utilize the memory hierarchy. I developed techniques based on *reuse distance (RD) analysis* for reasoning about parallel program memory behavior on multicore processors. In particular, I proposed techniques for scaling multicore RD profiles, enabling rapid prediction of memory performance for large-scale processors and problem sizes from profiles acquired on small-scale configurations. This dramatically reduces the time required to perform the analyses. In addition, I applied multicore RD analysis to identify optimal cache hierarchies across different applications, and to quantify the gains that cache hierarchy specialization can provide. Lastly, I extended multicore RD analysis to permit reasoning about directory cache behavior.

My ideas for multicore RD analysis were published in the *International Conference on Parallel Architectures and Compilation Techniques* (PACT) [C.16], the *International Symposium on Computer Architecture* (ISCA) [C.17], the *International Symposium on High Performance Computer Architecture* (HPCA) [C.19], and the *ACM Transactions on Computer Systems* (TOCS) [B.14, B.16]. Both ISCA and HPCA are top-tier conferences in the Computer Architecture field, with ISCA being *the* premier venue. My most recent work in this area has also been accepted for publication in a third *ACM TOCS* paper [B.17], and will appear in the near future. In addition, I have given an invited talk on multicore RD analysis at IBM T. J. Watson.

Approximate or Soft Computing. Many computations require precise and exact results. But this is overkill for computations in which users are satisfied with "good enough" results. Such *soft computations* afford designers the opportunity to trade off solution quality for improved performance, power, or reliability, providing a new dimension for optimizing systems. I proposed relaxed notions of correctness based on application-specific fidelity metrics to quantify solution quality. Using such definitions of *application-level correctness*, I conducted fault injection campaigns and demonstrated that the vast majority of single-bit event upsets occurring in soft computations do not alter solution quality appreciably even though architectural state is corrupted. Hence, systems that permit slight solution quality degradations can provide fault protection at much lower cost compared to traditional systems. To demonstrate, I designed and built fault recovery techniques that only checkpoint data critical to a program's correct execution–*i.e.*, its "hard state." By not protecting the "soft state," I showed significant savings in checkpointing overheads can be achieved. This work was published in the *International Symposium on High-Performance Computer Architecture* (HPCA) [C.14], with an extension of my HPCA paper appearing in the *Journal of Instruction-Level Parallelism* (JILP) [B.10]. Early versions of the work also appeared in refereed workshops [C.24, C.25].

When I first published this work, the idea of trading off solution quality to improve systems was unheard of in the computer architecture community. Today, there is an entire sub-discipline in architecture known as *approximate computing* formed around this idea, and it has an enormous following. Most approximate computing researchers are pursuing performance and power optimizations, though a sizable number are also looking at reliability. But all of them have embraced relaxing correctness for optimization purposes, and using fidelity metrics to assess the correctness impact of their techniques as proposed in my HPCA paper [C.14]. My research also had an immediate impact at the time I conducted it. The work was initiated during my sabbatical at the USC Information Sciences Institute (ISI) in 2005. During my sabbatical, I participated in a multi-PI project funded under Architectures for Cognitive Information Processing (ACIP), a major computer architecture program at DARPA. My ideas for soft computations became one of the central themes for the ISI team, and were presented to the director of DARPA (at the time, Dr. Anthony Tether) as an example of one of the most innovative directions coming out of the entire ACIP program.

Dynamic Resource Allocation. In multithreaded and multicore CPUs, a major problem is distribution of hardware resources across simultaneously executing threads for best performance and fairness. To address this problem, I invesetigated two dynamic resource allocation algorithms. First, I proposed *Learning-based Resource Allocation* for distributing pipeline resources in a simultaneous multithreading (SMT) processor. This technique treats resource allocation as a search problem, employing *hill-climbing* to intelligently explore the search space and identify the best resource allocation. Second, I also proposed *probabilistic replacement* (PR) for distributing shared cache resources in a multicore CPU. This technique identifies memory-intensive threads at runtime, and

"pushes back" on their allocation requests to ensure sufficient cache is left for non-memory-intensive threads.

My work on hill-climbing resource allocation was published in the *International Symposium on Computer Architecture* (ISCA) [C.13], with an extended version appearing in the *ACM Transactions on Computer Systems* (TOCS) [B.11]. Also, my PR work was published in the *International Symposium on Parallel Architectures and Compiler Techniques* (PACT) [C.15]. My hill-climbing technique is considered one of the state-of-the-art approaches for SMT processors, and has received significant attention. As of June 2016, there have been 110 citations on Google Scholar to my original ISCA paper introducing the idea. Of these 110 citing papers, 15 papers from 8 different research groups have independently implemented my hill-climbing technique, either to compare against or to use as the basis for further extension.

Helper Threads. In the past, several researchers have investigated using helper threads to *speculatively pre-execute* performance-degrading instructions in front of an application thread. I designed and built the first source-to-source compiler to extract code for pre-executing cache-missing loads in helper threads automatically, thus relieving the hardware or programmer of this onerous task. An interesting finding from this work is pre-execution code exhibits more parallelism than application code because problematic data and control dependences do not normally contribute to memory address computation, and can be removed. Hence, a compiler can orchestrate the execution of cache-missing loads in parallel using helper threads even though it is unable to parallelize the application code from which the cache-missing loads originate.

My work on compiler-based pre-execution has been published in the *International Conference on Architectural Support for Programming Languages and Operating Systems* (ASPLOS) [C.10] and the *International Symposium on Code Generation and optimization* (CGO) [C.11]. An extension of the former has been published in the *ACM Transactions on Computer Systems* (TOCS) [B.8]. Also, the foundational ideas for this work were recognized by an NSF CAREER Award. In addition, I have given invited talks on compiler-based pre-execution at Intel, Sun Microsystems, and IBM T. J. Watson, and a member of my research group worked at Intel's Microprocessor Research Laboratory to help develop a compiler for pre-execution. Finally, I gave an invited tutorial on this subject at the *International Conference on Parallel Architectures and Compilation Techniques* (PACT) along with Prof. Dean Tullsen from UC San Diego and Dr. Steve Liao from Intel, two other researchers who investigated helper thread techniques.

Pointer Prefetching. I proposed a new pointer prefetching technique, called *Multi-Chain Prefetching*, that addresses the *pointer-chasing problem* based on the following observation: pointer-chasing codes typically traverse independent pointer chains along different "arms" of multi-dimensional data structures. While each pointer chain traversal is serial, separate pointer chains can be pursued in parallel. Multi-Chain Prefetching identifies such *inter-chain memory parallelism*, and computes a static schedule of pointer prefetches that aggressively overlaps cache misses suffered along independent pointer chains. At runtime, a hardware prefetch engine uses the off-line prefetch schedule to guide prefetching.

This work was published in the *International Conference on Parallel Architectures and Compilation Techniques* (PACT) [C.8] (with an early version appearing in the *Memory Wall Problem Workshop* [C.6]), and influenced early research in pre-execution. In the *28th International Symposium on Computer Architecture* (ISCA-XXVIII), Chi-Keung Luk introduced *Software-Controlled Pre-Execution* in which he proposed using helper threads to traverse multiple independent pointer

chains simultaneously in software, much like Multi-Chain Prefetching does in hardware. Luk recognized the importance of exploiting inter-chain memory parallelism for effective pre-execution, and credited my Multi-Chain Prefetching work for having studied this basic idea. In his ISCA paper (see Section 2.2.1), Luk devotes an entire page to discuss Multi-Chain Prefetching, its advantages over previous pointer prefetching techniques, and how inter-chain memory parallelism can be gainfully exploited using helper threads.

Multiprocessor Clusters. My Ph.D. dissertation proposed and studied *Multigrain Shared Memory*. This research advocates building large-scale shared memory machines using small-scale multiprocessors as building blocks. I designed a shared memory protocol that couples page-based software shared memory with cache-coherent hardware shared memory, thus creating a layered shared memory model across a cluster of multiprocessors. I also built *MGS*, an implementation of my layered hardware/software shared memory protocol on the MIT Alewife machine.

This work was originally published in the *International Symposium on Computer Architecture* (ISCA) [C.3], and extensions of this work have been published in the *ACM Transactions on Computer Systems* [B.3] and the *International Conference on Supercomputing* (ICS) [C.5]. After my ISCA publication, several systems resembling MGS were reported in the literature, including *HLRC-SMP* from Princeton University, *Cashmere-2L* from the University of Rochester, *clustered Shasta* from Digital Western Research Laboratory, and *SoftFLASH* from Stanford University and Silicon Graphics. The papers describing these projects all appeared in major conferences, and all credited my work for having made a significant contribution to the area (for example, the Cashmere-2L work, published in the *Symposium on Operating Systems Principles*, credited MGS for being the first system to have a layered shared memory protocol).

3 Teaching and Advising

a. Courses Taught

Semester	Course Number	Course Title	Enrollment	Evaluation	Dept Average
Spring 16	ENEE 446	Digital Computer Design	28	3.56†	
Fall 15	ENEE 150	Intermediate Programming Concepts for Engineers	58	3.59†	
Spring 15	ENEE 150	Intermediate Programming Concepts for Engineers	47	3.68†	
Fall 14	ENEE 150	Intermediate Programming Concepts for Engineers	32	3.68†	
Spring 14	ENEE 150	Intermediate Programming Concepts for Engineers	44	3.84	3.33
Fall 13	ENEE 646	Computer Architecture	11	3.71	3.38
Spring 13	ENEE 150	Intermediate Programming Concepts for Engineers	45	3.60	3.17

†Due to switching to a new course evaluation system, the overall evaluation score during these semesters was only available per section. The evaluation scores reported here are averages of the per section scores. In addition, the department average score was not available at all.

Semester	Course Number	Course Title	Enroll-ment	Evaluation	Dept Average
Fall 12	ENEE 150	Intermediate Programming Concepts for Engineers	53	3.68	3.48
Spring 12	ENEE 150	Intermediate Programming Concepts for Engineers	47	3.51	3.04
Fall 11	ENEE 150	Intermediate Programming Concepts for Engineers	55	3.62	3.30
Spring 11	ENEE 150	Intermediate Programming Concepts for Engineers	43	3.65	3.25
Fall 10	ENEE 150	Intermediate Programming Concepts for Engineers	46	3.53	3.06
Spring 10	ENEE 150	Intermediate Programming Concepts for Engineers	32	3.71	3.39
Fall 09	ENEE 646	Computer Architecture	16	3.89	3.28
Spring 09	ENEE 114	Programming Concepts for Engineers	10	3.84	3.29
Fall 08	ENEE 114	Programming Concepts for Engineers	31	3.73*	3.18
Spring 08	ENEE 114	Programming Concepts for Engineers	79	3.58	3.39
Fall 07	ENEE 114	Programming Concepts for Engineers	36	3.11	3.03
Spring 07	ENEE 446	Digital Computer Design	14	3.48	3.17
Fall 06	ENEE 114	Programming Concepts for Engineers	49	3.25	3.36
Spring 06	ENEE 759A	Parallel Processing Computer Architecutres	13	3.68	3.47
Fall 04	ENEE 646	Computer Architecture	30	3.64	3.04
Spring 04	ENEE 446	Digital Computer Design	20	3.44	3.11
Fall 03	ENEE 759A	Parallel Processing Computer Architectures	20	3.59	3.45
Spring 03	ENEE 446	Digital Computer Design	44	3.40	3.19
Spring 03	ENEE 449	Undergraduate Research Project	2		
Fall 02	ENEE 446	Digital Computer Design	45	3.47	3.24
Spring 02	ENEE 446	Digital Computer Design	26	3.52	3.28
Fall 01	ENEE 446	Digital Computer Design	29	3.47	3.29
Spring 01	ENEE 446	Digital Computer Design	32	3.46	3.48
Fall 00	ENEE 759A	Parallel Processing Computer Architectures	21	3.84	3.52
Spring 00	ENEE 446	Digital Computer Design	23	3.63	3.40
Fall 99	ENEE 446	Digital Computer Design	37	3.64	3.37
Spring 99	ENEE 759A	Parallel Processing Computer Architectures	12	3.93	3.60
Fall 98	ENEE 646	Computer Architecture	59	4.58	4.28
Spring 98	ENEE 608B	Computer Engineering Seminar	15	N/A	N/A

*Highest rating in the department for 100-level (freshmen) courses, and second highest rating in the department overall.

b. Course or Curriculum Development

Since 2010, I have served as Director of Computer Engineering Education for the Electrical and Computer Engineering department at the University of Maryland. As the Computer Engineering director, I have been responsible for the curriculum that affects our undergraduate Computer Engineering majors (about 450 students) as well as our M.S. and Ph.D. students studying Computer Engineering (about 80 students). In particular, I have helped introduce two new undergraduate education programs at UMD, as well as a third program at the Naval Air Warfare Center–Aircraft Division (NAWCAD) in Southern Maryland. I have also overseen the introduction of six new courses into the CE curriculum at UMD. In addition, I have made major contributions to the courses that I personally teach.

More specifically, I have been involved in introducing the following education programs:

- **Informal Area of Concentration in Cybersecurity**. This program permits undergraduate CE majors to specialize in cybersecurity by taking 5 senior-level courses. Students receive commendation on their academic record after having completed the program. I was solely responsible for creating this progam, including design of its requirements, writing of the proposal, and securing approval from all required units (ECE and CS departments, ECE General Academic Affairs Committee, and campus Programs, Curricula, and Courses committee). This program became available to our students in Spring 2015.

- **Minor in Computer Engineering**. This program permits students to gain knowledge in Computer Engineering through a sequence of 7 courses. Any student at the University of Maryland who is not majoring in Computer Engineering can enroll in this program. The program was designed and developed by Professor Uzi Vishkin. As CE Director, I was involved in discussions on its requirements. In addition, I presented the program to the campus Programs, Curricula, and Courses committee to obtain final approval. This program became available to our students in Spring 2016.

- **STEM Workforce Development System for NAWCAD**. This program enhances exposure to STEM fields for both high school and college students in Southern Maryland, with the intent of expanding the workforce at the Naval Air Warfare Center–Aircraft Division (NAWCAD). It requires developing several STEM-related courses, and is a large collaboration between NAWCAD, St. Mary's County Public Schools, St. Mary's College of Maryland, the ECE and ME departments at the University of Maryland, and the College of Southern Maryland. Led by The Patuxent Partnership (TPP), we submitted a proposal to the Office of Naval Research to acquire funds for this program. The proposal was awarded in June 2015 (see G.18 under "Contracts and Grants" above). I served as the PI from the ECE department at the University of Maryland for this proposal.

I have also helped introduce the following undergraduate- and graduate-level courses into the CE curriculum at UMD. For each course, the faculty member(s) who taught the course were responsible for the primary course development. My role was to provide input on the course content, and to secure all necessary administrative approvals for each course.

- **ENEE 245–Digital Circuits and Systems Laboratory**. This is a fundamental sophomore-level digital logic design laboratory. In this lab course, students implement digital hardware using both discrete chips and FPGAs. And, they are introduced to hardware description languages (Verilog). It is required for all undergraduates in our department–*i.e.*, including both Computer Engineering and Electrical Engineering majors.

- **ENEE 447–Operating Systems**. Operating systems is a required course for all CE majors. Until recently, the only choice was the CS department's course, CMSC 412, which became a severe limitation as the CE major grew in size. ENEE 447 was a pre-existing operating systems course offered in the ECE department. It was substantially updated with several significant projects, and was then approved for the CE major. Today, CE majors can take either ENEE 447 or CMSC 412 to fulfill their operating systems requirement.

- **ENEE 459E–Topics in Computer Engineering: Introduction to Cryptology**. Until recently, cryptology was offered by the Computer Science and Math departments only. After hiring three faculty members in cybersecurity, the ECE department developed its own cryptology course. This is a senior-level course, and is an elective for both Computer Engineering and Electrical Engineering majors.

- **ENEE 640–VLSI Architecture**. This is a first-year graduate-level course on VLSI design. It is one of four core courses in the Computer Engineering technical area within the ECE department's graduate program.

- **ENEE 645–Compilers and Optimization**. This is a first-year graduate-level course on compilers. It is one of four core courses in the Computer Engineering technical area within the ECE department's graduate program.

- **ENEE 657–Computer Security**. This is an introductory graduate-level course on computer security. It is a non-core course that can be taken as an elective by our graduate students. It is offered regularly, once every 2 years.

Besides the contributions I have made to course and curriculum development as the CE Director, I have also made significant contributions to the courses that I personally teach. I was also involved in curriculum development efforts before becoming the CE Director. These contributions are described below:

- **ENEE 759A–Parallel Processing Computer Architectures**. I introduced an advanced topics course on parallel architectures in Spring 1999, and offered the course again in Fall 2000, Fall 2003, and Spring 2006. To enable projects in this course, I ported a simulator of a shared memory multiprocessor written at MIT to run on the University of Maryland UNIX workstations. I was the sole developer of this course.

- **ENEE 446–Digital Computer Design, and ENEE 646–Computer Architecture**. I revised the content of ENEE 446 and ENEE 646. In particular, I created new lecture notes for both courses. And, I developed several new programming projects, including three processor pipeline simulators and two processor cache simulators.

- **ENEE 150–Intermediate Programming Concepts for Engineers**. I created a new set of lecture notes for this course. In addition, I created 18 C programming projects for

students to gain programming experience. Each project consists of about 400 lines of code and a detailed project description that provides the project specification.

- **ENEE 114–Programming Concepts for Engineers**. I created a new set of lecture notes and programming projects for this course. (Some of the programming projects mentioned above for ENEE 150 were originally developed as part of ENEE 114).
- **Doctoral Program Revision**. I served on a committee of five graduate students that examined and recommended revisions to the requirements for the Ph.D. degree in Computer Science at the Massachusetts Institute of Technology. Summer 1996.
- **Electrical Engineering Undergraduate Curriculum Revision**. I served on a committee of several faculty that examined and recommended revisions to the entire undergraduate curriculum in Electrical Engineering at the University of Maryland. 1998 – 2000.

c. Manuals, Notes, Web Pages and Other Contributions to Teaching

- Lecture notes. As mentioned above, I created lecture notes for ENEE 114, ENEE 150, ENEE 446, ENEE 646, and ENEE 759A.
- Course web pages. I have developed web pages for all courses that I teach to publicize course information, and to disseminate lecture notes, assignments, and course announcements.

d. Teaching Awards and Other Special Recognition

- George Corcoran Memorial Award, May 2000. (Note, this award also listed in Section 2.h).

  "Presented to Professor Donald Yeung for significant contributions to Electrical and Computer Engineering Education, in recognition of teaching and education leadership at the College Park campus, effective contributions at the national level, and creative and other scholarly activities related to Electrical and Computer Engineering education."

- Eta Kappa Nu faculty inductee, May 2001.

e. Advising (non-resesarch)

I meet regularly with students to discuss graduate school, career advancement and opportunities, etc. Every year, I meet with several students in this capacity spread across both undergraduates and graduates.

i. Undergraduate

Eric Lee	January 2000
Kerry Kimes	December 2000
Fatemeh Shariatmadari	May 2001
Tia Gao	March 2002

Lisa Vora	April 2002
Paul Yu	August 2002
Eric Naber	October 2002
Frank Yang	January 2003
Patricia Kirsch	January 2003
Steve Tjoa	February 2003
Yogeeta Purohit	May 2003
Matthias Stearn	September 2004 – June 2008 (Inventis)
Patrick Valvidia	September 2004 – June 2008 (Inventis)
Michael Weston	September 2004 – June 2008 (Inventis)
Jared Wolinsky	September 2004 – June 2008 (Inventis)
Josh Davis	October 2007 – June 2011 (Inventis)
Kaushik Veeraraghavan	March 2005
Divya Jhalani	March 2005
Anqi Fu	April 2009
Jeremy Peggins	October 2010
Henrik Molintas	March 2011
George Sineriz	March 2011
David Shiao	September 2011
Mamadou Ba	December 2011
Rose Agger	January 2012
Emmeline Zhu	May 2012
Lida Ramsey	September 2012
Janani Gururam	February 2013
Kevin Chen	November 2013
Shiv Kaul	March 2015
Rebecca Gagnon	August 2015
Ben Hurwitz	March 2016

ii. Graduate

Zoltan Safar	May 1998 – present
Jayanth Gummaraju	December 2000
T Vinod Kumar Gupta	January 2001
Aneesh Aggarwal	November 2002 – 2003
Theodoros Salonidis	November 2002
Mohammed Zahran	November 2002
Renju Thomas	January 2003
Chungsoo Lim	April 2003
Michael Black	October 2004
Aamer Jaleel	November 2005
Aydin Balkan	December 2006
Vishal Khandelwal	December 2006
Azadeh Davoodi	December 2006
Brinda Ganesh	June 2007

Joseph Gross	April 2010
Paul Rosenfeld	August 2012
Elliott Cooper-Balis	June 2012
John Miller	November 2014

f. Advising (research)

i. Undergraduate

Aditi Sharma, Summer 1999
Project: *Irregular Data Streams in Data-Intensive Applications*
Eric Lee, Spring 2000
Project: *Measuring Chain-Depth in Pointer-Chasing Programs*
Robert Koberr, Summer 2000
Project: *Implementing Informing Memory Operations*
Chuck Tsen, Summer 2000
Project: *Implementing Informing Memory Operations*
Joe Sivak, Spring 2003
Project: *Extending Transparent Threading Mechanisms*
Hassan Chafi, Summer 2003
Project: *Thread Priority Mechanisms for Simultaneous Multithreaded Processors*
Dan Cardy, Summer 2003
Project: *Thread Priority Mechanisms for Simultaneous Multithreaded Processors*
Joshua Hatfield, Spring 2012
Project: *Efficient Reuse Time Profiling for On-line Reuse Distance Profile Prediction*
Elakian Kanakaraj, Fall 2013 – Spring 2014
Project: *Characterization of the MineBench Parallel Benchmarks*

ii. Master's

Ilya Khazanov, non-thesis option, Fall 1999
Project: *Irregular Data Streams in Data-Intensive Applications*
Rishi Kurichh, non-thesis option, Fall 1999
Project: *Irregular Data Streams in Data-Intensive Applications*
Employer: Myriad Logic
Changping Li, non-thesis option, Spring 2000
Project: *Parallelizing the Traveling Salesman Problem on Multigrain Systems*
Employer: Information Sciences Institute
Nicholas Kohout, thesis option, Spring 2000
Project: *Multi-Chain Prefetching*
Employer: Intel Corporation
Sujata Ramasubramanian, non-thesis option, Spring 2000
Project: *Characterizing the Performance of Irregular Applications*
Employer: Booz Allen Hamilton
Abdel-Hameed Abdel-Salem Badawy, thesis option, Spring 2002

Project: *Evaluating the Impact of Memory System Performance on Software Prefetching and Locality Optimizations*
Employer: N/A (continuing PhD)

Deepak Agarwal, thesis option, Spring 2002
Project: *Exploiting Application-Level Information to Reduce Memory Bandwidth Consumption*
Employer: Advanced Micro Devices

Gautham Dorai, thesis option, Summer 2002
Project: *Transparent Threads*
Employer: Google

Sumit Pamnani, thesis option, Summer 2003
Project: *Performance and Power Dissipation of Markov Prefetching*
Employer: Advanced Micro Devices

Priyanka Rajkhowa, thesis option, Summer 2006
Project: *Exploiting Soft Computing for Real Time Performance*

Xu Yang, non-thesis option, Spring 2011
Project: *Memory Profiling on the Tilera TILE64 Processor*

Lisa Stechschulte, thesis option, Spring 2012
Project: *Automatic Critical Section Discovery Using Memory Usage Patterns*
Employer: National Security Agency (employed prior to entering M.S. program)

Jeff Casarona, non-thesis option, Spring 2015
Project: *Using Reuse Distance Analysis to Identify Optimal Cache Hierarchies*
Employer: Northrop Grumman

Michael Zuzak, thesis option, Spring 2016
Project: *Exploiting Nested Parallelism on Heterogeneous Computing Systems*
Employer: U.S. Naval Research Laboratory

iii. Doctoral

Dongkeun Kim, Fall 2004
Project: *Compiler-Based Pre-Execution*
Employer: Samsung

Seungryul Choi, Spring 2006
Project: *Hill-Climbing SMT Processor Resource Distribution*
Employer: Google

Xuanhua Li, Spring 2009
Project: *Exploiting Inherent Program Redundancy for Fault Tolerance Enhancement*
Employer: Advanced Micro Devices

Wanli Liu, Spring 2009
Project: *Using Locality and Interleaving Information to Improve Shared Cache Performance*
Employer: National Institutes of Health

Meng-Ju Wu, Spring 2012
Project: *Reuse Distance Analysis for Large-Scale Chip Multiprocessors*

Employer: MathWorks

Abdel-Hameed Abdel-Salam Badawy, Summer 2013

Project: *Locality Transformations and Prediction Techniques for Optimizing Multicore Memory Performance*

Employer: Arkansas Tech

Inseok Choi, Spring 2014

Project: *Greedy Coordinate Descent Methods for CMP Multi-Level Cache Resizing*

Employer: Samsung

Minshu Zhao, Spring 2015

Project: *Studying the Impact of Multicore Processor Scaling on Cache Coherence Directories Using Reuse Distance Analysis*

Employer: MathWorks

Michael Badamo, expected Spring 2018

Project: *On-line Multicore Cache Hierarchy Adaptation*

Daniel Gerzhoy, expected Spring 2020

Project: *Cache Coherence for Heterogeneous Microprocessors*

g. Thesis Committees

Ph.D. Thesis Defense

Shamik Sharma, April 1998.
Ladan Gharai, July 1998.
Evan Rosser, September 1998.
Yuan-Shin Hwang, October 1998.
Zengyan Zhang, November 1998.
David Hellman, November 1998.
Hwansoo Han, January 2001.
Robert Bennett, October 2001.
Mohamed Zahran, February 2003.
Aneesh Aggarwal, March 2003.
Qingmin Shi, July 2003.
David Wang, April 2005.
Aamer Jaleel, November 2005.
Yu Deng, March 2006.
Sumesh Udayakumaran, May 2006.
Rania Mameesh, November 2006.
Kursad Albayraktaroglu, March 2007.
Michael Black, April 2007.
Brinda Ganesh, May 2007.
Sean Leventhal, April 2008.
Qin Wang, June 2008.
Aydin Balkan, July 2008.

Ph.D. Proposal Defense

Hwansoo Han, October 1999.
Aneesh Aggarwal, July 2001.
David Wang, February 2002.
Qingmin Shi, April 2002.
Sumesh Udayakumaran, August 2004
Michael Black, October 2004.
Aydin Balkan, December 2004.
Brinda Ganesh, October 2005.
Qin Wang, November 2006.
Sean Leventhal, November 2006.
Sangchul Song, May 2008.
George Caragea, May 2009.
Yufu Zhang, October 2009.
Mu-Tien Chang, July 2012.
Caleb Serafy, February 2014.
Tim Creech, January 2015.
Tiantao Lu, April 2015.
Kyungjin Yoo, December 2015.
Zhiyuan Yang, April 2017.

Cagdas Dirik, September 2009.
Sang Chul Song, August 2010.
Matthew Simpson, November 2010.
Elliott Cooper-Balis, April 2012.
Aparna Kotha, April 2013.
Bing Shi, May 2013.
Mu-Tien Chang, November 2013.
Khaled El Wazeer, February 2014.
Paul Rosenfeld, April 2014.
Kishan Sudusinghe, May 2015.
Tim Creech, August 2015.
Paul Tschirhart, September 2015.
Teng Long, December 2015.
JinSeong Jeon, February 2016.
Yufu Zhang, April 2016.
Caleb Serafy, May 2016.
Tiantao Lu, May 2016.

.

M.S. Thesis Defense

Efraim Berkovich, December 1998.
Theodorus Salonidas, April 1999.
Chao Wang, May 2000.
Chris Collins, May 2000.
Moussa Abdoul Ba, December 2000.
Dorit Naishlos, December 2000.
Mukul Khandelia, January 2001.
Kun Luo, March 2001.
Tiebing Zhang, April 2001.
Jayanth Gummaraju, May 2001.
Sumit Lohani, December 2001.
Aamer Jaleel, April 2002.
Zhang Yi, May 2002.
Michael Black, June 2003.
Barath Iyer, August 2003.
Hima Kakaraparthi, October 2003.
Chungsoo Lim, November 2003.
Lei Zhong, December 2003.
Pei Gu, April 2004.
Justin Teller, June 2004.
Jeff Scott Smith, December 2006.
Kiran Sheshadri, December 2007.
Joseph Glenn Gross, August 2010.

M.S. Scholarly Paper

Vippin Boyanapalli, May 1998.
Shashi Rao, April 1999.
Sujata Ramasubramanian, December 1999.
Jack Song, April 2000.
Yanfeng Mao, May 2000.
Changping Li, July 2000.
Gokturk Ozer, December 1999.
Mao Fang, Spring 2000.
Angelo Dominguez, Spring 2001.
Kaveri Pant, Spring 2001.
Mazen Kharbutli, April 2002.
Leila Rahmani, August 2002.
Vaddadi Chandu, August 2002.
Ata Karbasi, December 2002.
Amit Apte, April 2006.
Matt Davis, August 2006.
Austin Lanham, August 2007.
Chinmoy Gavini, December 2010.
Akshaya Nagendra, April 2016.

4 Service

I have been active in service, both within my profession as well as at the University of Maryland at College Park. In professional service, I have been an associate editor for the ACM Transactions on Architecture and Code Optimization (TACO). I have also served on the organizing committees, including the program committees and external review committees, of several conferences and workshops. In addition, I have been a seession chair at many conferences. And, I regularly serve as a reviewer for the top conferences and journals within my field. In section "a" below, I list these professional service activities in detail.

In addition to professional service, I have also been active in service to my University. Since 2010, I have been the Director of Computer Engineering Education during which time I have carried out many service-related functions (on top of my educational contributions described earlier). In particular, I have spoken at prospective freshmen recruitment events (about 8 per year), and conducted admissions for the minor in Computer Engineering. I have also evaluated numerous course equivalency and student petition requests. Moreover, I have also initiated and led meetings of the Computer Engineering faculty to discuss curricular issues. And, I am the liason between the ECE and CS faculty, meeting frequently with the CS associate chair for undergraduate studies to discuss curricular issues that affect Computer Engineering students. Outside of my duties as the Computer Engineering director, I have also served on numerous committees at the department, college, and university levels. In section "b" below, I list these University service activies in detail.

a. Professional

- Panel Member. *National Science Foundation Major Research Instrumentation Program.* March 2001.
- Program Committee Member. *34th International Symposium on Microarchitecture.* August 2001.
- Session Chair. *34th International Symposium on Microarchitecture.* December 2001.
- Program Committee Member. *6th Workshop on Languages, Compilers, and Run-time Systems for Scalable Computers.* February 2002.
- Session Chair. *6th Workshop on Languages, Compilers, and Run-time Systems.* March 2002.
- Organizer. *Tutorial on Architecture and Compiler Support for Speculative Precomputation*, held in conjunction with the *12th Annual International Conference on Parallel Architectures and Compilation Techniques.* September 2003. (With Prof. Dean Tullsen from UC San Diego and Dr. Steve Liao from Intel).
- Program Committee Member. *3rd International Symposium on Code Generation and Optimization with Special Emphasis on Feedback-Directed and Runtime Optimization.* October 2004.
- Finance Chair. *3rd International Symposium on Code Generation and Optimization with Special Emphasis on Feedback-Directed and Runtime Optimization.* October 2004.

- Session Chair. *3rd International Symposium on Code Generation and Optimization with Special Emphasis on Feedback-Directed and Runtime Optimization.* March 2005.
- Associate Editor. *ACM Transactions on Architecture and Code Optimization.* March 2008 – Present.
- Workshops and Tutorials Co-Chair. *41st International Symposium on Microarchitecture.* November 2008.
- Program Committee Member. *5th International Conference on High-Performance and Embedded Architectures and Compilers.* January 2010.
- Program Committee Member. *ACM International Conference on Computing Frontiers.* March 2012.
- Journal-First Program Selection Member (as part of ACM TACO). *8th International Conference on High-Performance and Embedded Architectures and Compilers.* January 2013.
- Journal-First Program Selection Member (as part of ACM TACO). *9th International Conference on High-Performance and Embedded Architectures and Compilers.* January 2014.
- Journal-First Program Selection Member (as part of ACM TACO). *10th International Conference on High-Performance and Embedded Architectures and Compilers.* January 2015.
- External Review Committee member. *42nd International Symposium on Computer Architecture.* January 2015.
- External Review Committee member. *48th International Symposium on Microarchitecture.* July 2015.
- Journal-First Program Selection Member (as part of ACM TACO). *11th International Conference on High-Performance and Embedded Architectures and Compilers.* January 2016.
- External Review Committee member. *49th International Symposium on Microarchitecture.* May 2016.
- Program Committee Member. *31st IEEE International Parallel and Distributed Processing Symposium.* January 2017.
- Referee for the following journals:

  ACM Transactions on Architecture and Code Optimization
  ACM Transactions on Computer Systems
  Computer Journal
  Proceedings of the IEEE
  IEEE Computer
  IEEE Computer Society Press
  IEEE Micro
  IEEE Transactions on Computers
  IEEE Transactions on Parallel and Distributed Systems
  Journal on Parallel and Distributed Computing

- Referee for the following conferences:

International Conference on Architectural Support for Programming Languages and Operating Systems
International Conference on Measurement and Modeling of Computer Systems
International Conference on Parallel Architectures and Compilation Techniques
International Conference on Parallel Processing
International Conference on Supercomputing
International Parallel and Distributed Processing Symposium
International Parallel Processing Symposium
International Symposium on Code Generation and Optimization
International Symposium on Computer Architecture
International Symposium on High-Performance Computer Architecture
International Symposium on Microarchitecture
International Symposium on Parallel Architectures, Algorithms and Networks
Supercomputing

b. University

i. Director of Computer Engineering Education

- Attended and spoke at around 8 prospective freshman recruitment events per year.
- Conducted admissions for the minor in Computer Engineering program.
- Evaluated courses taken by our students while at other Universities for equivalency against our own courses in the ECE and CS departments.
- Leader for meetings of the Computer Engineering faculty to discuss curricular issues.
- Liason between ECE and CS faculty for Computer Engineering curricular issues.
- Met frequently with the CS associate chair for undergraduate studies to discuss curricular issues affecting Computer Engineering students.

ii. Department Committees

- Member. *ECE Curriculum Revision Committee*. 1998 – 2000.
- Member. *ECE Department Council*. 1998 – 2002.
- Member. *ECE Computer Staff Hiring Committee*. 1999.
- Member. *Department Plan Review Adhoc Committee*. May 2000.
- Member. *Emeritus Status Criteria Adhoc Committee*. October 2000.
- Member. *Computer Engineering Curriculum Committee*. 2000.
- Appointed Member. *Graduate Studies and Research Committee*. 2002.

- Member. *Graduate Studies and Research Committee.* 2003.
- Member. *Computer Engineering Curriculum Revision Committee.* 2003.
- Chair. *Graduate Studies and Research Committee.* 2004.
- Member. *ECE Department Council.* 2004.
- Member. *Undergraduate Affairs Committee.* 2004.
- Member. *Computer Engineering Curriculum Oversight Committee.* 2004 – 2005.
- Member. *Faculty Hiring Committee.* 2005 – 2007.
- Chair. *ECE Department Council.* 2006 – 2007.
- Member. *ECE Department Council.* 2007 – 2008.
- Member. *General Academic Affairs Committee.* 2006 – 2010.
- Member. *Research Coordinator Hiring Committee.* 2008.
- Member. *Future Faculty Program (FFP) Selection Committee.* 2010.
- Member. *Distinguished Dissertation Fellowship (DDF) Selection Committee.* 2010.
- Member. *Undergradaute Affairs Committee.* 2013 – 2015.
- Member. *Faculty Scholarship Committee.* 2014.
- Member. *Salary Committee.* 2014 – present.
- Member. *Facilities and Services Committee.* 2015 – present.
- Co-Chair. *Undergraduate Affairs Committee.* 2016 – present.

iii. College Committees

- Member. *Pelczar and Phi Delta Gamma Awards Committee.* March 1998.
- Member. *Articulation Committee.* 1998.

iv. University Committees

- Member. *University Medal Selection Committee.* April 2011.

c. Presentations and Media Activities

- Interview regarding rising enrollments in Computer Engineering programs. International Data Group (IDG) / Network World. July 2012.
- Interview with Donald Yeung - Associate Professor of Electrical and Computer Engineering at the University of Maryland. EEWeb. http://www.eeweb.com/spotlight/interview-with-dr.-donald-yeung. February 2016.