

Junfeng Yang

Computer Science Department
Columbia University
460 CSB, Mail Code 0401
New York, NY 10027

Phone: (212) 939-7012
Fax: (212) 666-0140
junfeng@cs.columbia.edu
<http://www.cs.columbia.edu/~junfeng>

Research Interests

Operating systems, software reliability, concurrency, and security

Education

2008	Ph.D. in Computer Science <i>Dissertation: Automatically Finding Serious Storage System Errors</i> <i>Advisor: Dawson Engler</i>	STANFORD UNIVERSITY
2002	M.S. in Computer Science	STANFORD UNIVERSITY
2000	B.S. in Computer Science <i>Dissertation: Computer-aided Rapid Prototyping of Mechanical Parts</i> Entrance exam waived based on outstanding high-school record	TSINGHUA UNIVERSITY, CHINA
1995-1997	Special academic program Only 60 freshmen selected	TSINGHUA UNIVERSITY, CHINA

Professional Experience

2014-2018	Co-founder and President	NIMBLEDROID, INC. (ACQUIRED BY HEADSPIN)
2013-now	Associate Professor	COLUMBIA UNIVERSITY
2013-2016	EMC-endowed Chair Professor Group (Adjunct)	TSINGHUA UNIVERSITY
2008-2013	Assistant Professor	COLUMBIA UNIVERSITY
2011-now	Co-director, Software Systems Lab	COLUMBIA UNIVERSITY
2008-2011	Director, Reliable Computer Systems Lab	COLUMBIA UNIVERSITY
2008-2012	Technical Consultant	MICROSOFT RESEARCH, SILICON VALLEY
2007-2008	Postdoc Researcher Received the "exceeded" (best) rating, rarely given to postdocs or first-year hires	MICROSOFT RESEARCH, SILICON VALLEY
2001-2007	Research Assistant	STANFORD UNIVERSITY
06/2005-09/2005	Research Intern	MICROSOFT RESEARCH, SILICON VALLEY

Honors and Awards

2017	Best Paper Award of the 26th ACM Symposium on Operating Systems Principles (SOSP)
2014	Google Faculty Research Award
2012	Sloan Research Fellowship
2012	Air Force Office of Scientific Research Young Investigator Research Program (AFSOR YIP) Award
2011	National Science Foundation Faculty Early Career (NSF CAREER) Award

2004	Best Paper Award of the Sixth USENIX Symposium on Operating Systems Design and Implementation (OSDI)
2003	Stanford School of Engineering Fellowship
2002	Siebel Scholarship
1995–1997	Tsinghua Outstanding Scholarships for Undergraduates
1995	Chinese Mathematical Olympiad Winter Camp
1994	Chinese Mathematical Olympiad, First Prize
1994	Chinese Physics Olympiad, First Prize, Hebei Province

Research Grants and Contracts (Total: \$31,887,487)

03/2019–02/2020	PI, J.P. Morgan Faculty Research Award, \$150,000, <i>Efficient Formal Safety Analysis of Neural Networks</i> , with co-PI: Jeannette M. Wing, Suman Jana
01/2019–12/2019	sole PI, DiDi Faculty Research Award, \$178,640, <i>Effective Testing and Verification of Deep Neural Networks</i>
09/2017–08/2020	co-PI, ONR N00014-17-1-2788, \$3,243,244, <i>ABIDES: Adaptive Binary Debloating and Security</i> , with Georgios Portokalidis (PI), Vasileios P. Kemerlis
02/2017–01/2018	sole PI, Canon, \$75,000, <i>Reliability and Security of Cyber-Physical Systems</i>
11/2016–3/2017	sole PI, Huawei, \$50,000, <i>Validation of Distributed System Using Model Checking</i>
09/2016–08/2020	PI, NSF CNS-1564055, \$1,200,000, <i>Efficient Repair of Learning Systems via Machine Unlearning</i> , with co-PI: Yinzhi Cao
03/2016–02/2019	PI, ONR N00014-16-1-2263, \$1,114,488, <i>Adapting Static and Dynamic Program Analysis to Effectively Harden Debloated Software</i> , with co-PI: Georgios Portokalidis
03/2016–02/2017	sole PI, WeChat, \$34,285, <i>Availability of Large Scale Distributed Systems</i>
02/2016–01/2017	sole PI, Canon, \$75,000, <i>Reliability and Security of Cyber-Physical Systems</i>
08/2015–07/2018	co-PI, ONR, \$1,415,127, <i>The YOLO Approach to Resilient Cyber-Physical Systems</i> , with Simha Sethumadhavan (PI)
02/2014–01/2015	sole PI, Google, \$77,807, <i>Efficiently, Effectively Detecting Mobile App Bugs with AppDoctor</i>
06/2013–09/2013	sole PI, NSF CNS-1340511, \$16,000, <i>Research Experiences for Undergraduates (REU) for LOOM: a Language and System for Bypassing and Diagnosing Concurrency Errors</i>
06/2013–09/2013	sole PI, NSF CNS-1340506, \$16,000, <i>Research Experiences for Undergraduates (REU) for Guanyin: a Thousand hands with a Thousand eyes for Distributed Software Checking</i>
09/2012–08/2016	co-PI, NSF CCF-1162021, \$800,000, <i>SHF: Medium: RacePro: Automatically Detecting API Races in Deployed Systems</i> , with Jason Nieh (PI)
2012–2016	sole PI, Sloan Foundation, \$50,000, <i>Sloan Research Fellowship</i>
07/2012–06/2016	sole PI, AFOSR YIP, \$359,998, <i>Concurrency Attacks and Defenses</i>
07/2012–06/2016	PI, ONR N00014-12-1-0166, \$749,975, <i>Transparently Extending Programs at Compilation to Prevent Bugs</i> , with co-PI: Angelos Keromytis
09/2011–01/2016	co-PI, DARPA MRC, \$6,619,270, <i>MEERKATS: Maintaining Enterprise Resiliency via Kaleidoscopic Adaptation and Transformation of Software Services</i> , with Angelos Keromytis (PI), Roxana Geambasu, Simha Sethumadhavan, Sal Stolfo
09/2011–08/2014	sole PI, NSF CNS-1117805, \$250,000, <i>LOOM: a Language and System for Bypassing and Diagnosing Concurrency Errors</i>

- 02/2011–01/2017 sole PI, NSF CAREER CNS-1054906, \$450,000, *CAREER: Making Threads More Deterministic by Memoizing Schedules*
- 10/2010–08/2015 co-PI, DARPA CRASH, \$6,424,180, *SPARCHS: Symbiotic, Polymorphic, Autonomic, Resilient, Clean-slate, Host Security*, with Simha Sethumadhavan (PI), Sal Stolfo, Angelos Keromytis, David August
- 07/2010–06/2013 PI, NSF CNS-1012633, \$325,000, *SemGrep: a System for Improving Software Reliability through Semantic Similarity Bug Search*, with co-PI: Angelos Keromytis, Dawson Engler
- 08/2010–01/2015 co-PI, IARPA, \$7,530,113, *MINESTRONE*, with Angelos Keromytis (PI), Sal Stolfo
- 09/2009–08/2014 PI, NSF CNS-0905246, \$1,012,000, *Guanyin: a Thousand hands with a Thousand eyes for Distributed Software Checking*, with co-PI: Gail Kaiser, Jason Nieh

Publications and Talks

(papers available online at <http://www.cs.columbia.edu/~junfeng>)

Refereed Conference Publications

- [1] Dongdong She, Kexin Pei, Dave Epstein, Junfeng Yang, Baishakhi Ray, and Suman Jana. NEUZZ: Efficient fuzzing with neural program smoothing. In *Proceedings of the 2019 IEEE Symposium on Security and Privacy (S&P '19)*, 2019.
- [2] Miguel A. Arroyo, M. Tarek Ibn Ziad, Hidenori Kobayashi, Junfeng Yang, and Simha Sethumadhavan. YOLO: Frequently resetting cyber-physical systems for security. In *SPIE Defense and Commercial Sensing*, 2019.
- [3] Shiqi Wang, Kexin Pei, Justin Whitehouse, Junfeng Yang, and Suman Jana. Efficient formal safety analysis of neural networks. In *Proceedings of the Thirty-second Annual Conference on Neural Information Processing Systems (NIPS)*, 2018.
- [4] Gang Hu, Linjie Zhu, and Junfeng Yang. Appflow: Using machine learning to synthesize robust, reusable ui tests. In *Proceedings of the 2018 11th Joint Meeting on Foundations of Software Engineering, FSE '18*, 2018.
- [5] Shiqi Wang, Kexin Pei, Justin Whitehouse, Junfeng Yang, and Suman Jana. Formal security analysis of neural networks using symbolic intervals. In *Proceedings of the 27th USENIX Security Symposium*, 2018.
- [6] Xinhao Yuan, Junfeng Yang, and Ronghui Gu. Partial order aware concurrency sampling. In *Proceedings of the 30th International Conference On Computer Aided Verification (CAV '18)*, 2018.
- [7] Shixiong Zhao, Rui Gu, Haoran Qiu, Tsz On Li, Yuexuan Wang, Heming Cui, and Junfeng Yang. OWL: Understanding and detecting concurrency attacks. In *IEEE/IFIP International Conference on Dependable Systems and Networks (DSN '18)*. IEEE, 2018.
- [8] Yinzhi Cao, Alexander Fangxiao Yu, Andrew Aday, Eric Stahl, Jon Merwine, and Junfeng Yang. Efficient repair of polluted machine learning systems via causal unlearning. In *Proceedings of the 13th ACM ASIA Conference on Information, Computer and Communications Security*, 2018.
- [9] Kexin Pei, Yinzhi Cao, Junfeng Yang, and Suman Jana. Deepxplore: Automated whitebox testing of deep learning systems. In *Proceedings of the 26th ACM Symposium on Operating Systems Principles (SOSP '17)*, October 2017. (acceptance rate: 16.8%, 39/232) **Won Best Paper Award**. Also appeared at MLSec '17
- [10] David Williams-King, Graham Gobieski, Kent Williams-King, James P Blake, Xinhao Yuan, Patrick Colp, Vasileios P Kemerlis, Junfeng Yang, and William Aiello. Shuffler: Fast and deployable continuous code re-randomization. In *Proceedings of the Twelfth Symposium on Operating Systems Design and Implementation (OSDI '16)*, 2016. (acceptance rate: 18.1%, 47/260)
- [11] Yang Tang, Gang Hu, Xinhao Yuan, Lingmei Weng, and Junfeng Yang. Grandet: A unified, economical object store for web applications. In *Proceedings of the Seventh ACM Symposium on Cloud Computing, SoCC '16*, 2016. (acceptance rate: 25.2%, 38/151)
- [12] Eric Koskinen and Junfeng Yang. Reducing crash recoverability to reachability. In *Proceedings of the 39th Annual Symposium on Principles of Programming Languages (POPL '16)*, pages 267–283, January 2016. (acceptance rate: 23.3%, 59/253)
- [13] Heming Cui, Rui Gu, Cheng Liu, Tianyu Chen, and Junfeng Yang. Paxos made transparent. In *Proceedings of the 25th ACM Symposium on Operating Systems Principles (SOSP '15)*, October 2015. (acceptance rate: 16.1%, 30/186)

- [14] Heming Cui, Rui Gu, Cheng Liu, and Junfeng Yang. Repframe: An efficient and transparent framework for dynamic program analysis. In *Proceedings of 6th Asia-Pacific Workshop on Systems (APSys '15)*, July 2015.
- [15] Yang Tang and Junfeng Yang. Secure deduplication of general computations. In *Proceedings of the USENIX Annual Technical Conference (USENIX ATC '15)*, 2015. (acceptance rate: 15.8%, 35/221)
- [16] Suzanna Schmeelk, Junfeng Yang, and Alfred Aho. Android malware static analysis techniques. In *The 10th Annual Cyber and Information Security Research (CISR) Conference*, 2015.
- [17] Yinzhi Cao and Junfeng Yang. Towards making systems forget with machine unlearning. In *Proceedings of the 2015 IEEE Symposium on Security and Privacy (S&P '15)*, 2015. (acceptance rate: 13.5%, 55/407)
- [18] Gang Hu, Xinhao Yuan, Yang Tang, and Junfeng Yang. Efficiently, effectively detecting mobile app bugs with appdoctor. In *Proceedings of the 2014 ACM European Conference on Computer Systems (EUROSYS '14)*, April 2014. (acceptance rate: 18.4%, 27/147)
- [19] Heming Cui, Jiri Simsa, Yi-Hong Lin, Hao Li, Ben Blum, Xinan Xu, Junfeng Yang, Garth A. Gibson, and Randal E. Bryant. Parrot: a practical runtime for deterministic, stable, and reliable threads. In *Proceedings of the 24th ACM Symposium on Operating Systems Principles (SOSP '13)*, November 2013. (acceptance rate: 18.8%, 30/160)
- [20] Jingyue Wu, Gang Hu, Yang Tang, and Junfeng Yang. Effective dynamic detection of alias analysis errors. In *Proceedings of the Ninth Joint Meeting of the European Software Engineering Conference and the ACM SIGSOFT International Symposium on Foundations of Software Engineering (ESEC-FSE '13)*, August 2013. (acceptance rate: 20.3%, 51/251)
- [21] Heming Cui, Gang Hu, Jingyue Wu, and Junfeng Yang. Verifying systems rules using rule-directed symbolic execution. In *Eighteenth International Conference on Architecture Support for Programming Languages and Operating Systems (ASPLOS '13)*, 2013. (acceptance rate: 23.1%, 44/191)
- [22] Jingyue Wu, Yang Tang, Gang Hu, Heming Cui, and Junfeng Yang. Sound and precise analysis of parallel programs through schedule specialization. In *Proceedings of the ACM SIGPLAN 2012 Conference on Programming Language Design and Implementation (PLDI '12)*, pages 205–216, June 2012. (acceptance rate: 18.8%, 48/255)
- [23] Heming Cui, Jingyue Wu, John Gallagher, Huayang Guo, and Junfeng Yang. Efficient deterministic multithreading through schedule relaxation. In *Proceedings of the 23rd ACM Symposium on Operating Systems Principles (SOSP '11)*, pages 337–351, October 2011. (acceptance rate: 18.3%, 28/153)
- [24] Oren Laadan, Nicolas Viennot, Chia che Tsai, Chris Blinn, Junfeng Yang, and Jason Nieh. Pervasive detection of process races in deployed systems. In *Proceedings of the 23rd ACM Symposium on Operating Systems Principles (SOSP '11)*, October 2011. (acceptance rate: 18.3%, 28/153)
- [25] Huayang Guo, Ming Wu, Lidong Zhou, Gang Hu, Junfeng Yang, and Lintao Zhang. Practical software model checking via dynamic interface reduction. In *Proceedings of the 23rd ACM Symposium on Operating Systems Principles (SOSP '11)*, pages 265–278, October 2011. (acceptance rate: 18.3%, 28/153)
- [26] Ding Yuan, Yinglian Xie, Rina Panigrahy, Junfeng Yang, Chad Verbowski, and Arunvijay Kumar. Context-based online configuration-error detection. In *Proceedings of the USENIX Annual Technical Conference (USENIX '11)*, 2011. (acceptance rate: 15.0%, 27/180 full paper)
- [27] Heming Cui, Jingyue Wu, Chia-Che Tsai, and Junfeng Yang. Stable deterministic multithreading through schedule memoization. In *Proceedings of the Ninth Symposium on Operating Systems Design and Implementation (OSDI '10)*, October 2010. (acceptance rate: 16.1%, 32/199)

- [28] Jingyue Wu, Heming Cui, and Junfeng Yang. Bypassing races in live applications with execution filters. In *Proceedings of the Ninth Symposium on Operating Systems Design and Implementation (OSDI '10)*, October 2010. (acceptance rate: 16.1%, 32/199)
- [29] Mark Gabel, Junfeng Yang, Yuan Yu, Moises Goldszmidt, and Zhendong Su. Scalable and systematic detection of buggy inconsistencies in source code. In *Conference on Object-Oriented Programming Systems, Languages, and Applications (OOPSLA '10)*, October 2010. (acceptance rate: 27.4%, 45/164)
- [30] Junfeng Yang, Tisheng Chen, Ming Wu, Zhilei Xu, Xuezheng Liu, Haoxiang Lin, Mao Yang, Fan Long, Lintao Zhang, and Lidong Zhou. MODIST: Transparent model checking of unmodified distributed systems. In *Proceedings of the Sixth Symposium on Networked Systems Design and Implementation (NSDI '09)*, pages 213–228, April 2009. (acceptance rate: 19.6%, 32/163)
- [31] Junfeng Yang, Can Sar, and Dawson Engler. Explode: a lightweight, general system for finding serious storage system errors. In *Proceedings of the Seventh Symposium on Operating Systems Design and Implementation (OSDI '06)*, pages 131–146, November 2006. (acceptance rate: 18.0%, 27/150)
- [32] Junfeng Yang, Can Sar, Paul Twohey, Cristian Cadar, and Dawson Engler. Automatically generating malicious disks using symbolic execution. In *Proceedings of the 2006 IEEE Symposium on Security and Privacy (S&P '06)*, pages 243–257, May 2006. (acceptance rate: 9.2%, 23/251)
- [33] Junfeng Yang, Paul Twohey, Dawson Engler, and Madanlal Musuvathi. Using model checking to find serious file system errors. In *Proceedings of the Sixth Symposium on Operating Systems Design and Implementation (OSDI '04)*, pages 273–288, December 2004. (acceptance rate: 14.0%, 27/193) **Won Best Paper Award.**
- [34] Ted Kremenek, Ken Ashcraft, Junfeng Yang, and Dawson Engler. Correlation exploitation in error ranking. In *Proceedings of the 12th ACM SIGSOFT International Symposium on Foundations of Software Engineering (SIGSOFT '04/FSE-12)*, pages 83–93, November 2004. (acceptance rate: 14.8%, 25/169)
- [35] Junfeng Yang, Ted Kremenek, Yichen Xie, and Dawson Engler. MECA: an extensible, expressive system and language for statically checking security properties. In *Proceedings of the 10th ACM conference on Computer and communications security (CCS '03)*, pages 321–334, October 2003. (acceptance rate: 13.8%, 35/253)
- [36] Andy Chou, Junfeng Yang, Benjamin Chelf, Seth Hallem, and Dawson Engler. An empirical study of operating systems errors. In *Proceedings of the 18th ACM Symposium on Operating Systems Principles (SOSP '01)*, pages 73–88, November 2001. (acceptance rate: 20.0%, 17/85)

Refereed Workshop Publications

- [37] Xinhao Yuan, David Williams-King, Junfeng Yang, and Simha Sethumadhavan. Making lock-free data structures verifiable with artificial transactions. In *Eighth Workshop on Programming Languages and Operating Systems (PLOS '15)*, October 2015.
- [38] Junfeng Yang, Heming Cui, and Jingyue Wu. Determinism is overrated: What really makes multithreaded programs hard to get right and what can be done about it? In *the Fifth USENIX Workshop on Hot Topics in Parallelism (HOTPAR '13)*, June 2013.
- [39] Junfeng Yang, Ang Cui, Sal Stolfo, and Simha Sethumadhavan. Concurrency attacks. In *the Fourth USENIX Workshop on Hot Topics in Parallelism (HOTPAR '12)*, June 2012.
- [40] Angelos Keromytis, Roxana Geambasu, Simha Sethumadhavan, Sal Stolfo, Junfeng Yang, Azzedine Benameur, Marc Dacier, Matthew Elder, Darrell Kienzle, and Angelos Stavrou. The MEERKATS cloud security architecture. In *Proceedings of the third International Workshop on Security and Privacy in Cloud Computing (ICDCS-SPCC '12)*, June 2012.

- [41] Simha Sethumadhavan, Salvatore Stolfo, Angelos Keromytis, Junfeng Yang, and David August. The SPARCHS project: Hardware support for software security. In *Proceedings of the First SysSec Workshop (SYSSEC '11)*, 2011.
- [42] Angelos Keromytis, Salvatore Stolfo, Junfeng Yang, Angelos Stavrou, Anup Ghosh, Dawson Engler, Marc Dacier, Matthew Elder, and Darrell Kienzle. The MINESTRONE architecture: Combining static and dynamic analysis techniques for software security. In *Proceedings of the First SysSec Workshop (SYSSEC '11)*, 2011.
- [43] Oren Laadan, Chia che Tsai, Nicolas Viennot, Chris Blinn, Peter Senyao Du, Junfeng Yang, and Jason Nieh. Finding concurrency errors in sequential code—os-level, in-vivo model checking of process races. In *Proceedings of the 131th USENIX workshop on Hot topics in operating systems (HOTOS '11)*, 2011. (acceptance rate: 24.8%, 33/133)
- [44] Qifa Ke, Vijayan Prabhakaran, Yinglian Xie, Yuan Yu, Jingyue Wu, and Junfeng Yang. Optimizing data partitioning for data-parallel computing. In *Proceedings of the 131th USENIX workshop on Hot topics in operating systems (HOTOS '11)*, 2011. (acceptance rate: 24.8%, 33/133)
- [45] Junfeng Yang, Paul Twohey, Ben Pfaff, Can Sar, and Dawson Engler. eXplode: A lightweight, general approach for finding serious errors in storage systems. In *Proceedings of the first Workshop on the Evaluation of Software Defect Detection Tools (BUGS '05)*, June 2005.

Refereed Journal Publications

- [46] Kexin Pei, Yinzhi Cao, Junfeng Yang, and Suman Jana. Deepxplore: Automated whitebox testing of deep learning systems. *Communications of the ACM*, 2019. **(invited)**
- [47] Kexin Pei, Yinzhi Cao, Junfeng Yang, and Suman Jana. Deepxplore: Automated whitebox testing of deep learning systems. *GetMobile: Mobile Comp. and Comm.*, 22(3):36–38, January 2019. **(invited)**
- [48] Junfeng Yang, Heming Cui, Jingyue Wu, Yang Tang, and Gang Hu. Determinism is not enough: Making parallel programs reliable with stable multithreading. *Communications of the ACM*, 2014. **(invited)**
- [49] Heming Cui, Gang Hu, Jingyue Wu, and Junfeng Yang. Verifying systems rules using rule-directed symbolic execution. *SIGARCH Comput. Archit. News*, 41(1):329–432, 2013. (also listed in Refereed Conference Publications)
- [50] Jingyue Wu, Yang Tang, Gang Hu, Heming Cui, and Junfeng Yang. Sound and precise analysis of parallel programs through schedule specialization. *SIGPLAN Not.*, 47(6):205–216, June 2012. (also listed in Refereed Conference Publications)
- [51] Mark Gabel, Junfeng Yang, Yuan Yu, Moises Goldszmidt, and Zhendong Su. Scalable and systematic detection of buggy inconsistencies in source code. *SIGPLAN Not.*, 45(10):175–190, October 2010. (also listed in Refereed Conference Publications)
- [52] John MacCormick, Nicholas Murphy, Venugopalan Ramasubramanian, Udi Wieder, Junfeng Yang, and Lidong Zhou. Kinesis: A new approach to replica placement in distributed storage systems. *ACM Transactions on Storage Systems*, 4(4):1–28, January 2009.
- [53] Ted Kremenek, Ken Ashcraft, Junfeng Yang, and Dawson Engler. Correlation exploitation in error ranking. *SIGSOFT Softw. Eng. Notes*, 29(6):83–93, October 2004. (also listed in Refereed Conference Publications)
- [54] Junfeng Yang, Paul Twohey, Dawson Engler, and Madanlal Musuvathi. Using model checking to find serious file system errors. *ACM Transactions on Computer Systems*, 24(4):393–423, November 2006. **(invited)**
- [55] Andy Chou, Junfeng Yang, Benjamin Chelf, Seth Hallem, and Dawson Engler. An empirical study of operating systems errors. *SIGOPS Oper. Syst. Rev.*, 35(5):73–88, October 2001. (also listed in Refereed Conference Publications)

Technical Reports

- [56] Xinhao Yuan, David Williams-King, Junfeng Yang, and Simha Sethumadhavan. Making lock-free data structures verifiable with artificial transactions. Technical Report CUCS-026-14, Columbia University.
- [57] Jingyue Wu, Gang Hu, Yang Tang, and Junfeng Yang. Effective dynamic detection of alias analysis errors. Technical Report CUCS-003-13, Columbia University.
- [58] Junfeng Yang, Heming Cui, Jingyue Wu, John Gallagher, Chia che Tsai, Huayang Guo, Yang Tang, and Gang Hu. Make parallel programs reliable with stable multithreading. Technical Report CUCS-006-13, Columbia University.
- [59] Junfeng Yang, Ang Cui, John Gallagher, Sal Stolfo, and Simha Sethumadhavan. Concurrency attacks. Technical Report CUCS-028-11, Columbia University, 2011.
- [60] Jingyue Wu, Heming Cui, and Junfeng Yang. Bypassing races in live applications with execution filters. Technical Report CUCS-036-10, Columbia University.
- [61] Nageswar Keetha, Leon Wu, Gail Kaiser, and Junfeng Yang. Distributed eXplode: A high-performance model checking engine to scale up state-space coverage. Technical Report CUCS-051-08, CS Department, Columbia University, December 2008.
- [62] Nageswar Keetha, Leon Wu, Gail Kaiser, and Junfeng Yang. A software checking framework using distributed model checking and checkpoint/resume of virtualized process domains. Technical Report CUCS-032-09, CS Department, Columbia University, June 2009.

Patents

- Junfeng Yang, Lintao Zhang, Lidong Zhou, Zhenyu Guo, Xuezheng Liu, Jian Tang, Mao Yang, *Distributed system checker*, US Patent 7984332,
- Rina Panigrahy, Chad Verbowski, Yinglian Xie, Junfeng Yang, Ding Yuan, *Fingerprinting event logs for system management troubleshooting*, US Patent 8069374

Invited Talks

- 5/2019 “Artificial Intelligence for Good: Thoughts, Deeds and Words” panelist. Moderator: Andrew W. Smyth
- 10/2018 Data Innovation Network “How Deep Learning Can Go Wrong: Use cases to drive discussion on technical and business implications with the use of deep learning” session moderator.
- 10/2018 “Taming the AI Fire.” Department of Defense CIO Meeting, Columbia Data Science Institute, Host: Columbia Data Science Institute Director Jeannette M. Wing and Columbia Engineering Sr. Executive Vice Dean
- 10/2018 “Taming the AI Fire.” Board of Visitors Meeting, Columbia Engineering. Host: Dean Mary Boyce and Sr. Executive Vice Dean
- 5/2018 “Tools 2.0: Towards Effective Tools to Build Safer and More Secure Machine Learning Systems.” Microsoft Research Asia. Host: Lintao Zhang
- 5/2018 “Tools 2.0: Towards Effective Tools to Build Safer and More Secure Machine Learning Systems.” Peking University. Host: Yao Guo
- 5/2018 “Tools 2.0: Towards Effective Tools to Build Safer and More Secure Machine Learning Systems.” Tsinghua University. Host: Yong Cui
- 3/2018 “Tools 2.0: Towards Effective Tools to Build Safer and More Secure Machine Learning Systems.” Distinguished lecture, Purdue. Host: Yiyang Zhang
- 3/2018 “Tools 2.0: Towards Effective Tools to Build Safer and More Secure Machine Learning Systems.” Data Science Day, Columbia. Host: Jeannette M. Wing

- 10/2017 "Practical Model Checking of Real-world Distributed Systems." Cloud Reliability Workshop. Host: Nemat Bidokhti
- 4/2017 "AI for Software Testing." QCon Beijing. Host: Wenxin Huang
- 7/2016 "Need for Speed: Software Tools Edition." Microsoft Research Faculty Summit. Host: Suman Nath
- 3/2016 "Build Performant Apps: Metrics, Common Issues, and Best Practices." Droidcon San Francisco. Host: Apps4all, Touchlab
- 12/2015 "Build Fluid Apps with Android Profiling Tools." AnDevCon Santa Clara. Host: BZ Media
- 7/2015 "Build Fluid Apps with Android Profiling Tools." AnDevCon Boston. Host: BZ Media
- 4/2014 "Determinism Is Not Enough: Making Parallel Programs Reliable with Stable Multithreading." Princeton University. Host: Michael Freedman
- 4/2014 "Determinism Is Not Enough: Making Parallel Programs Reliable with Stable Multithreading." University of Washington. Host: Tom Anderson
- 02/2014 "Determinism Is Not Enough: Making Parallel Programs Reliable with Stable Multithreading." UCLA. Host: Todd Millstein
- 12/2013 "Determinism Is Not Enough: Making Parallel Programs Reliable with Stable Multithreading." UT Austin. Host: Emmett Witchel
- 11/2013 "Parrot: A Practical Runtime for Deterministic, Stable, and Reliable Threads." Princeton University. Host: Michael Freedman
- 11/2013 "Determinism Is Not Enough: Making Parallel Programs Reliable with Stable Multithreading." UCSD. Host: Yuanyuan Zhou
- 11/2013 "Determinism Is Not Enough: Making Parallel Programs Reliable with Stable Multithreading." Stanford University. Host: Alex Aiken
- 10/2013 "Determinism Is Not Enough: Making Parallel Programs Reliable with Stable Multithreading." Cornell University. Host: Andrew Myers
- 08/2013 "Determinism Is Not Enough: Making Parallel Programs Reliable with Stable Multithreading." Microsoft Research Asia. Host: Lintao Zhang
- 08/2013 "Determinism Is Not Enough: Making Parallel Programs Reliable with Stable Multithreading." Beijing University. Host: Yao Guo
- 08/2013 "Determinism Is Not Enough: Making Parallel Programs Reliable with Stable Multithreading." Shanghai Jiaotong University. Host: Haibo Chen
- 07/2013 "Effectively Model Check Real-World Distributed Systems." National University of Singapore. Host: Jin Song Dong
- 06/2013 "How Useful Is Determinism for Reliability?" Invited panel "Determinism: Blessing or Curse" at Fifth USENIX Workshop on Hot Topics in Parallelism. Host: Emery Berger and Kim Hazelwood
- 04/2013 "Effectively Model Check Real-World Distributed Systems." Rutgers. Host: Santosh Narakatte
- 12/2012 "Effectively Model Check Real-World Distributed Systems." CMU. Host: Garth Gibson
- 10/2012 "Pervasive Detection of Process Races in Deployed Systems." University of Southern California. Host: Minlan Yu
- 06/2012 "Improving the Reliability and Security of Parallel Programs." Tsinghua University. Host: Wenguang Chen

- 06/2012 "Improving the Reliability and Security of Parallel Programs." Beihang University. Host: Chunming Hu
- 06/2012 "Improving the Reliability and Security of Parallel Programs." Beijing University. Host: Yao Guo
- 06/2012 "Efficiently and Stably Making Threads Deterministic." Invited talk at 4th International Workshop on Practical Synthesis (co-located with PLDI). Host: Martin Vechev and Eran Yahav
- 12/2011 "Efficiently and Stably Making Threads Deterministic." Microsoft Research. Host: Madan Musuvathi
- 11/2011 "Efficiently and Stably Making Threads Deterministic." Telefonica Research at Spain. Host: Michael Sirivianos
- 06/2011 "Efficiently and Stably Making Threads Deterministic." Duke. Host: Landon Cox
- 06/2011 "Efficiently and Stably Making Threads Deterministic." NCSU. Host: Xiaohui Helen Gu
- 12/2010 "Techniques to Defend Against Concurrency Attacks." Columbia S&P day. Host: Angelos Keromytis
- 11/2010 "Making Threads Deterministic by Memoizing Schedules." Yale. Host: Bryan Ford
- 11/2010 "Making Threads Deterministic by Memoizing Schedules." CMU. Host: Garth Gibson
- 10/2010 "Making Threads Deterministic by Memoizing Schedules." IBM Watson. Host: Dilma Da Silva
- 08/2009 "MODIST: Transparent Model Checking of Unmodified Distributed Systems." Stony Brook University. Host: R. Sekar
- 12/2008 "Automatically Generating Malicious Disks using Symbolic Execution." IBM S&P day. Host: Vugranam Sreedhar
- 06/2006 "EXPLODE: a Lightweight, General System for finding Serious Storage System Errors." Microsoft Research, Silicon Valley. Host: Lidong Zhou
- 10/2006 "Automatically Finding Serious Storage System Errors." Coverity, San Francisco, CA. Host: Andy Chou
- 09/2005 "Graceful Degradation." Microsoft Research, Silicon Valley. Host: Lidong Zhou
- 03/2005 "Automatically Finding Serious File System Errors." Stanford Computer Forum Security Workshop, Stanford, CA.
- 05/2005 "FiSC, an Effective File System Checker." Stanford Networking Research Center, Stanford, CA.

Conference and Workshop Talks

- 06/2013 "Determinism Is Not Enough: What Really Makes Multithreaded Programs Hard to Get Right, and What Can Be Done about It?" *Fifth USENIX Workshop on Hot Topics in Parallelism*, San Jose, California
- 06/2012 "Concurrency Attacks." *Fourth USENIX Workshop on Hot Topics in Parallelism*, Berkeley, California
- 04/2009 "MODIST: Transparent Model Checking of Unmodified Distributed Systems." *Sixth USENIX Symposium on Networked Systems Design and Implementation (NSDI '09)*, Boston, Massachusetts
- 11/2006 "EXPLODE: a Lightweight, General System for finding Serious Storage System Errors." *Seventh USENIX Symposium on Operating Systems Design and Implementation (OSDI '06)*, Seattle, Washington

- 05/2006 "Automatically Generating Malicious Disks using Symbolic Execution." *IEEE Symposium on Security and Privacy (S&P '06)*, Oakland, California
- 12/2004 "Using Model Checking to Find Serious File System Errors." *Sixth USENIX Symposium on Operating Systems Design and Implementation (OSDI '04)*, San Francisco, California
- 10/2003 "MECA: an Extensible, Expressive System and Language for Statically Checking Security Properties." *Tenth ACM conference on Computer and communication security (CCS '03)*, Washington, DC

Professional Activities

Conference Program Chairmanship

2013 Co-chair, Program committee, *4th ACM SIGOPS Asia-Pacific Workshop on Systems (APSys '13)*

Conference Program Committees

2019 Member, Program committee, *27th Symposium on Operating System Principles (SOSP '19)*

2017 Member, Program committee, *26th Symposium on Operating System Principles (SOSP '17)*

2017 Member, Program committee, *38th Annual SIGPLAN Conference on Programming Language Design and Implementation (PLDI '17)*

2016 Member, Program committee, *37th IEEE Symposium on Security and Privacy (S&P) (S&P '16)*

2014 Member, Program committee, *11th USENIX Symposium on Operating Systems Design and Implementation (OSDI '14)*

2012 Member, Program committee, *10th USENIX Symposium on Operating Systems Design and Implementation (OSDI '12)*

2012 Member, Program committee, *10th USENIX Conference on File and Storage Technologies (FAST '12)*

2011 Member, Program committee, *2nd ACM SIGOPS Asia-Pacific Workshop on Systems (APSys '11)*

2011 Member, Program committee, *2011 USENIX Annual Technical Conference (USENIX ATC '11)*

2010 Member, Program committee, *9th USENIX Symposium on Operating Systems Design and Implementation (OSDI '10)*

2010 Member, Program committee, *5th International Workshop on Systems Software Verification (SSV '10)*

2010 Member, Program committee, *International Symposium on Engineering Secure Software and Systems (ESSoS '10)*

2007 Member, Program committee, *Workshop on Storage Security and Survivability (StorageSS '07)*

Conference Organization Committees

2014-now Steering Committee, *ACM SIGOPS Asia-Pacific Workshop on Systems (APSys)*

2013 Poster-Demo session program committee, *10th USENIX Symposium on Networked Systems Design and Implementation (NSDI '13)*

2012 Poster session co-chair, *10th USENIX Symposium on Operating Systems Design and Implementation (OSDI '12)*

2011 Publicity co-chair, *23rd ACM Symposium on Operating Systems Principles (SOSP '11)*

Journal and Conference Refereeing (in addition to conference program committees)

Annual Conference of the Special Interest Group on Data Communication (SIGCOMM)
USENIX Symposium on Operating Systems Design and Implementation (OSDI)
USENIX Conference on File and Storage Technologies (FAST)
USENIX Symposium on Networked Systems Design and Implementation (NSDI)

ACM Transactions on Computer Systems (TOCS)

ACM Transactions on Storage (TOS)

International Symposium on Distributed Computing (DISC)

IEEE Symposium on Security and Privacy (S&P)

IEEE Pervasive Computing

IEEE Workshop on Mobile Cloud Computing (MobiCC)

Funding Agency

Air Force Office of Scientific Research (AFOSR) Proposal Reviewer

Department of Energy (DOE) Panelist

National Science Foundation (NSF) Panelist, Proposal Reviewer

Professional Societies

Member, Association for Computing Machinery (ACM).

Member, Advanced Computing Systems Association (USENIX).

Member, Institute of Electrical and Electronic Engineers (IEEE).

University Activities

(all activities at Columbia University unless otherwise noted)

Ph.D. Students under Supervision

2018-2023 (expected) Chengzhi Mao
2016-2021 (expected) Kexin Pei
2014-2020 (expected) Lingmei Weng
2014-2020 (expected) David William-King
2012-2019 (expected) Xinhao Yuan
2011-2019 (expected) Yang Tang
2011-2018 Gang Hu
2009-2015 Heming Cui
2008-2014 Jingyue Wu

Ph.D. Dissertation Committees

2018 Gang Hu, "Techniques for Efficient and Effective Mobile Testing"
2017 Nipun Arora, "Live Debugging: Debugging Applications On-the-fly"
2016 Jonathan Bell, "Making Software More Reliable by Uncovering Hidden Dependencies"
2016 Nicolas Viennot, "Record-Replay Mechanisms Applied to Operating Systems and Distributed Systems," Ph.D. in Computer Science
2015 Vasileios P. Kemerlis, "Protecting Commodity Operating Systems through Strong Kernel Isolation," Ph.D. in Computer Science
2014 Heming Cui, "Stable Multithreading: A New Paradigm for Reliable and Secure Threads", Ph.D. in Computer Science
2014 Jingyue Wu, "Sound and Precise Analysis of Multithreaded Programs through Schedule Specialization", Ph.D. in Computer Science
2014 Jiri Simsa, "Systematic Testing of Concurrent Programs", Ph.D. in Computer Science, CARNEGIE MELLON UNIVERSITY
2013 Tan Tian Huat, "Verification and Analysis of Web Services Composition", Ph.D. in Computer Science, NATIONAL UNIVERSITY OF SINGAPORE
2011 Dinesh Subhraveti, "Record and vPlay: Problem Determination with Virtual Replay Across Heterogeneous Systems", Ph.D. in Computer Science
2011 Omer Boyaci, "High-Performance Multimedia Collaboration Tools for Application Sharing, Measuring Capture-to-display Latency, and User Created Services", Ph.D. in Computer Science
2010 Chris Murphy, "Metamorphic Testing Techniques to Detect Defects in Applications without Test Oracles", Ph.D. in Computer Science
2009 Wei Xu, "Program Transformation Techniques for Automatic Runtime Detection of Software Exploits", Ph.D. in Computer Science, STATE UNIVERSITY OF NEW YORK AT STONY BROOK

Ph.D. Dissertation Proposal Committees

04/2018 Xinhao Yuan, "Effective Detection and Prevention of Concurrency Bugs"

06/2017	Gang Hu, "Automating UI testing by synthesizing reusable, robust tests"
06/2017	Naser AlDuaij, "Multi-Mobile Computing"
11/2015	Nipun Arora, "Live Debugging: Debugging Applications On-the-fly"
11/2015	Jonathan Bell, "Faster, More Reliable Software Builds"
06/2015	Vasileios P. Kemerlis, "Building Secure Operating Systems"
01/2015	Nicolas Viennot, "Record-Replay Mechanisms Applied to Operating Systems and Distributed Systems"
09/2014	Heming Cui, "Stable Multithreading: A New Paradigm for Reliable and Secure Threads"
09/2013	Jingyue Wu, "Sound and Precise Analysis of Multithreaded Programs through Schedule Specialization"
04/2011	Jiri Simsa, "Systematic Testing of Concurrent Programs, Ph.D. Computer Science," CARNEGIE MELLON UNIVERSITY
03/2010	Omer Boyaci, "Multimedia Tools for Application Sharing, Measuring Capture-to-display Latency, and User Created Services"
01/2009	Dinesh Subhraveti, "Record and Transplay: Partial Checkpointing for Fault Determination"
12/2008	Chris Murphy, "Using Metamorphic Testing at Runtime to Detect Defects in Applications without Test Oracles"

Ph.D. Candidacy Exam Committees

03/2018	David William-King
05/2017	Lingmei Weng
11/2016	Xinhao Yuan
06/2016	Gang Hu
04/2016	Marios Pomonis
03/2016	George Argyros
12/2015	Theofilos Petsios
05/2015	David Tagatac
05/2015	Naser AlDuaij
11/2013	Nipun Arora
11/2013	Jonathan Bell
07/2013	Heming Cui
05/2013	Nicolas Viennot
12/2012	Vasileios P. Kemerlis
05/2011	Jingyue Wu
02/2010	Jong Yul Kim
04/2009	Leon Wu

M.S. Dissertation Committees

04/2017	Rui Gu, "Reliable Synchronization on Multithreaded Servers"
---------	---

Undergraduate Dissertation Supervision

2004-2006 Informally advised a Stanford Undergraduate Honor's thesis by Can Sar

High School Students Supervised for Research

2014 Andrew Aday, GUNN HIGH SCHOOL
2014 Ben Iofel, BERGEN COUNTY ACADEMIES
2013 Mustafa Ansari, PLAINVIEW-OLD BETHPAGE JFK HIGH SCHOOL
2013 Matthew Piccolella, POINT PLEASANT HIGH SCHOOL

Other Students and Scientists Supervised for Research

2018-now Nicholas James Bullard, undergraduate student
2018-now Graham Philip Patterson, undergraduate student
2018-now Brandon Zhang, undergraduate student
2018-now Rahul Kataria, undergraduate student
2018 Xinyuan Zhang, M.S. student
2018 Yi Zhang, M.S. student
2015-2018 Hidenori Kobayashi, Visiting Associate Research Scientist
2016-2019 Linjie Zhu, M.S. student
2014-2016 Rui Gu, M.S. student
2014-2016 James P. Blake, M.S. student
2014-2016 Graham Gobieski, undergraduate student
2014-2016 Michelle Zheng, undergraduate student
2014-2015 Karthik Jayaraman, M.S. student
2014-2015 Vishal Nandkishor Vyas, M.S. student
2014-2015 Yinzhi Cao, Postdoc Research Scientist
2014-2015 Shunkun Yang, Visiting Associate Research Scientist
2013-2014 Yanting Zhao, M.S. student
2014-2015 Xin Lu, M.S. student
2014 Patrick Yuen, REU student
2014 Jonathan Beekman, REU student
2013-2015 Yan Cui, Postdoc Research Scientist
2013 Alex Roth, REU student
2013 Addisu Petros, REU student
2012-2013 Hao Li, Postdoc Research Scientist
2011-2012 Junyang Lu, M.S. student
2011-2012 Chuliang Weng, Visiting Associate Research Scientist
2011 Oren Laadan, Postdoc Research Scientist, co-supervised with Jason Nieh
2011-2013 Younghoon Jeon, M.S. student
2011-2013 YiHong Lin, M.S. student

2010-2011	John Gallagher, M.S. student
2010-2011	Chia-che Tsai, M.S. student
2010-2011	Yunling Wang, M.S. student
2010-2011	Chris Blinn, undergraduate student

Departmental Service

2009-now	Facilities Committee
2011	Faculty Search Committee
2011	Ph.D. Admissions Committee
2009-now	Student Award Committee
2009-now	Ph.D. Comprehensive Examiner, Operating Systems
2012-now	Academic Committee
2011-now	M.S. Academic Adviser
2008-now	M.S. Application Reviewer. Reviewed over 40 applications in 2012, 49 in 2011, 26 in 2009, and 13 in 2008

Curriculum and Teaching

(all activities at Columbia University unless otherwise noted)

New Curriculum Development

- 2008-now Developed and taught a new graduate-level course on software reliability. Integrated research from four different fields: operating systems, security, programming languages, and software engineering. Designed programming labs and research projects aimed to teach students how to solve real-world problems leveraging real-world tools such as LLVM and Pin.
- 2011-now Developed and taught a new advanced undergraduate/graduate course in operating systems using MIT's xv6 teaching operating system.
- 2009-2010 Developed and taught an advanced undergraduate/graduate course in operating systems with new course materials and kernel programming assignments aimed to raise reliability-awareness in students. Incorporated current operating system research including some of my own into the lectures.

Teaching Experience

- Fall 2008-now Three project courses COMS E6901, W4901, and W3998. Total enrollment over three project courses: 6 (Spring 2019), 6 (Fall 2018), 2 (Spring 2018), 4 (Fall 2017), 4 (Fall 2016), 6 (Spring 2016), 6 (Fall 2015), 4 (Spring 2015), 9 (Fall 2014), 1 (Summer 2014), 4 (Spring 2014), 8 (Fall 2013), 6 (Spring 2013), 13 (Fall 2012), 6 (Spring 2012), 9 (Fall 2011), 4 (Spring 2011), 6 (Fall 2010), 3 (Spring 2010), 8 (Fall 2009), 3 (Spring 2009), 2 (Fall 2008)
- Spring 2018 COMS E6121 Reliable Software. Enrollment 14.
- Spring 2018 COMS E6898 Security and Robustness of Machine Learning. Enrollment 14.
- Spring 2018 COMS E6121 Reliable Software. Enrollment 6.
- Spring 2018 COMS E6898 Security and Robustness of Machine Learning. Enrollment 13.
- Spring 2017 COMS E6121 Reliable Software. Enrollment 8, instructor rating 4.83/5.0 (mean), 5/5.0 (median).
- Spring 2015 COMS W4118 Operating Systems I. Enrollment 3 (via CVN)
- Summer 2015 COMS W4118 Operating Systems I. Enrollment 2 (via CVN)
- Fall 2014 COMS E6121 Reliable Software. Enrollment 16, instructor rating 3.63/5.0 (mean), 3.30/5.0 (median).
- Fall 2013 COMS W4118 Operating Systems I. Enrollment 125, instructor rating 4.04/5.0 (mean), 4/5 (median).
- Fall 2012 COMS E6121 Reliable Software. Enrollment 12, instructor rating 4.0/5.0 (mean), 4/5 (median).
- Spring 2012 COMS W4118 Operating Systems I. Enrollment 56, instructor rating 4.1/5.0 (mean), 4/5 (median).
- Fall 2011 COMS E6121 Reliable Software. Enrollment 17, instructor rating 3.4/5.0 (mean), 3.5/5 (median).
- Spring 2011 COMS W4118 Operating Systems I. Enrollment 66, instructor rating 4.2/5.0 (mean), 4/5 (median).
- Fall 2010 COMS E6121 Reliable Software. Enrollment 11, instructor rating 5.0/5.0 (mean), 5/5 (median).
- Spring 2010 COMS W4118 Operating Systems I. Enrollment 56, instructor rating 3.7/5.0 (mean), 4/5 (median).

Fall 2009	COMS E6998 Reliable Software. Enrollment 13, instructor rating 3.9/5.0 (mean), 4/5 (median).
Spring 2019	COMS W4118 Operating Systems I. Enrollment 45, instructor rating 3.5/5.0 (mean), 4/5 (median).
Fall 2008	COMS E6998 How to Make Reliable Software. Enrollment 9, instructor rating 4.0/5.0 (mean), 4/5 (median).
2004-2005	Teaching Assistant, CS 240 Advanced Operating Systems

STANFORD UNIVERSITY

Selected Student Course Evaluation Comments

Fall 2013	<p>"I found Professor Yang to be extremely knowledgeable and clear in his delivery of the material. He was also extremely accommodating during office hours and never hesitated to give any extra instruction due time. I learned a lot from his instruction alone, and he's one of the best professors I've had in the CS department."</p> <p>"The class was worth more than the money I spent on it."</p> <p>"This course is well organized and a lot of effort has been made to include up-to-date contents and proper exercises."</p> <p>"Junfeng is an awesome teacher!"</p>
Spring 2012	<p>"Kudos! Courses like these justify the pain and cost of graduate studies."</p> <p>"Really committed professor."</p> <p>"Also it's very hard to set the difficulty level of this course, and I think Prof. Yang is very close to optimal by catering to a wide spectrum of student backgrounds."</p> <p>"The professor is dedicated to course preparation and anything that might help student understand the course well."</p>
Spring 2011	<p>"Prof. Yang is able to synthesize all the relevant information and teach it to the students in a way that is clear and expresses a hierarchy of ideas, making the material easy to review later as well. I felt like I couldn't afford to miss a single lecture because they were all so useful."</p> <p>"The switch to xv6 is a huge win for this course."</p> <p>"EXCELLENT knowledge of material, very well-prepared."</p> <p>"The professor has immense knowledge and experience in the subject."</p> <p>"Great class!"</p> <p>"Great course content and very lucid lectures."</p>
Fall 2010	<p>"Prof. Yang's ability to engage students to think critically about the state of the art papers in systems research made this course one of the most interesting of all I have taken at Columbia."</p> <p>"I really enjoyed this course. It was a unique opportunity to look at the state of the art in an area and critically analyze it."</p> <p>"Junfeng always had a beyond-thorough understanding of the material, and even when we didn't, he did a fantastic job of bringing it down to our level."</p> <p>"Professor Yang really cares about the development of students. He usually gives us very inspiring ideas, suggesting possible direction for starting on our researches. I am glad to have a chance to join his class."</p> <p>"Great class. Learned a lot about reliable software."</p>
Spring 2010	<p>"He never seemed stumped by a question, and always gave useful answers. The course was enlightening as well, and very interesting. I was excited to attend each course."</p>

“Great understanding of the topic. Never stumped by questions, and always gave useful answers to students’ questions.”

“Great course and great professor.”

“The professor had an amazing understanding of the topic at hand, and the sourcecode of the Linux kernel.”

Spring 2009

“Prof. Yang did an excellent job!”

Fall 2008

“knowledgable and helpful professor. Motivates his students and helps them realize their potential.”