Tongping Liu

CONTACT INFORMATION

tongping.cs@gmail.com

https://people.umass.edu/tongping/

(413)-695-2437

Dept. Electrical and Computer Engineering

University of Massachusetts Amherst

Amherst, MA

RESEARCH INTERESTS:

AI and Systems, Parallel Computing, Performance Analysis, Automatic Debugging, Operating/Runtime System

EDUCATION BACKGROUND

Ph.D., Computer Science, University of Massachusetts Amherst, May 2014.

> Thesis: Safe and Efficient Multithreading

> Advisor: Emery D. Berger

PROFESSIONAL EXPERIENCE

Assistant Professor, Department of Electrical and Computer Engineering, Univ. of Massachusetts Amherst, September 2019 - Now.

Assistant Professor, Department of Computer Science, Univ. of Texas at San Antonio, August 2014 - August 2019.

MAJOR AWARDS AND MEDIA REPORT

Watcher is reported by UMass News, Jiqizhixin, Sina, sohu, 163.com, kknews, thepaper, matpool, 51cto, linkerresearcher,......

NSF Scalable Parallelism in the Extreme (SPX) Award - 2018

CRII Award - 2016

Google Faculty Award - 2015

PROFESSIONAL SERVICE

Program Committee: SC'22, PPoPP'22, HPDC'22, ICDCS' 21, HPDC'21, LCTES'21, ICICS'21, HPDC'20, BigData'20, LCTES'20, COMPSAC'19, ISMM'19, TrustCom'18, SETTA '18, ICSDE'17, ICPADS'16, ICCCN'16, ICCCN'15,

External Review Committee: PLDI 2019, PPOPP 2015

Program Organizer: Publicity Chair ICDCS'21

Reviewer: IEEE Transactions on Software Engineering, TACO, Journal of Systems and Software, Journal of Computational Science, NCS17, CCGrid 2017, TDSC, TPDS, IJICT, TCC, TC, ICDCS 2013, Middleware 2013, ICDCS 2014

REPRESENTATIVE PUBLICATIONS (E.G. SOSP, PLDI, EUROSYS, CCS, MICRO, ICSE...)

Underlined names are students supervised by me.

PUBLICATIONS (E.G. SOSP, PLDI, EUROSYS, CCS, MICRO, ICSE...)

Underlined names are students supervised by me.

1. *UnHang: Deadlock Prediction via Generalized Dependency.*
   Jinpeng Zhou, Hanmei Yang, John Lange, Tongping Liu.
   The 2022 International Symposium on Software Testing and Analysis (ISSTA'22). Acceptance Rate: 24.4% (61/250).

2. *CachePerf: A Unified Cache Miss Classifier via Hybrid Hardware Sampling.*
   Jin Zhou, Steven Tang, Hanmei Yang, Tongping Liu.
   The 2022 ACM SIGMETRICS/Performance conference (**SIGMETRICS'22**). Acceptance Rate: 20.6% (21/102).

3. *FreeLunch: Compression-based GPU MemoryManagement for Deep Neural Networks.*
   Shaurya Patel, Hui Guang, Tongping Liu.
   Workshop on Memory Centric High Performance Computing (**MCHPC'21**)

4. *Dryadic: Flexible and Fast Graph Pattern Matching at Scale.*
   Daniel Mawhirter, Sam Reinehr, Wei Han, Noah Fields, Miles Claver, Connor Holmes, Jedidiah McClurg, Tongping Liu, Bo Wu.
   The 30th International Conference on Parallel Architectures and Compilation Techniques (**PACT'21**), pp. 289-303.

5. *GraphZero: A High-Performance Subgraph Matching System.*
   Daniel Mawhirter, Sam Reinehr, Connor Holmes, Tongping Liu, Bo Wu
   ACM SIGOPS Operating Systems Review, Volume 55, pages 21-37.

6. *NumaPerf: Predictive NUMA Profiling.*
   Xin Zhao, Jin Zhou, Hui Guan, Wei Wang, Xu Liu, Tongping Liu.
   Proceedings of The 35th ACM International Conference on Supercomputing (**ICS'21**), pp. 52-62. Acceptance Rate: 24.2% (38/157).

7. *Watcher: In-Situ Failure Diagnosis for In-Production Software.*
   Hongyu Liu, Sam Silvestro, Xiangyu Zhang, Jian Huang, Tongping Liu.
   Proceedings of the ACM on Programming Languages (**OOPSLA'20**). Acceptance Rate: 36% (109/302).

8. *Prober: Practically Defending Overflows with Page Protection.*
   Hongyu Liu*, Ruiqin Tian*, Bin Ren, Tongping Liu. [*: equally-contributed]
   Proceedings of the 35th IEEE/ACM International Conference on Automated Software Engineering (**ASE'20**). Acceptance Rate: 22.5% (93/408).

9. *CSOD: Context-sensitive Overflow Detection.*
   Hongyu Liu, Sam Silvestro, Xiaoyin Wang, Lide Duan, Tongping Liu.
   Proceedings of the 2019 IEEE/ACM International Symposium on Code Generation and Optimization ((**CGO'19**). Acceptance Rate: 31% (28/69).

10. *iReplyer: In-situ and Identical Record-and-Replay for Multithreaded Applications*
    Hongyu Liu, Sam Silvestro, Wei Wang, Chen Tian, Tongping Liu.
    Proceedings of The 37th annual ACM SIGPLAN conference on Programming Language Design and Implementation (**PLDI'18**). Acceptance Rate: 19.8% (55/277).

11. *Sampler: PMU-based Sampling to Detect Memory Errors Latent in Production Software*
    Sam Silvestro, Hongyu Liu, Tong Zhang, Changhee Jung, Dongyoon Lee, Tongping Liu.

To appear in Proceedings of The 51th International Symposium on Microarchitecture (**Micro'18**). Acceptance Rate: 21.1%.

12. *Guarder: A Tunable Secure Allocator*
Sam Silvestro, Hongyu Liu, Tianyi Liu, Zhiqiang Lin, Tongping Liu.
Proceedings of The 27th USENIX Security Symposium (**Security'18**). Acceptance Rate: 19.1% (100/524).

13. *A User Space-based Project for Practicing Core Memory Management Concepts*
Sam Silvestro, Timothy T. Yuen, Corey Crosser, Dakai Zhu, Turgay Korkmaz, Tongping Liu.
Proceedings of The 49th ACM Technical Symposium on Computer Science Education (**SIGCSE'18**).

14. *FreeGuard: A Faster Secure Heap Allocator*
Sam Silvestro, Hongyu Liu, Corey Crosser, Zhiqiang Lin, Tongping Liu.
ACM Conference on Computer and Communications Security (**CCS'17**). Acceptance Rate: 18% (151/836).

15. *UnDead: Defeating Deadlocks of Production Software*
Jinpeng Zhou, Sam Silvestro, Hongyu Liu, Yan Cai, and Tongping Liu. The 32nd IEEE/ACM International Conference on Automated Software Engineering (**ASE'17**). Acceptance Rate: 21% (65/314).

16. *SyncPerf: Categorizing, Detecting, and Diagnosing Synchronization Performance Bugs*
Mejbah ul Alam*, Tongping Liu*, Guangming Zeng, Abduallah Muzahid. [*: equally-contributed]
The 2017 European Conference on Computer Systems (**EuroSys'17**). Acceptance Rate: 20.5% (41/200).

17. *DoubleTake: Fast and Precise Error Detection via Evidence-Based Dynamic Analysis*
Tongping Liu, Charlie Curtsinger, Emery D. Berger.
The 38th International Conference on Software Engineering (**ICSE'16**). Acceptance Rate: 19% (101/530).

18. *Cheetah: Detecting False Sharing Efficiently and Effectively*.
Tongping Liu*, Xu Liu*. [*: equally-contributed]
International Symposium on Code Generation and Optimization (**CGO'16**). Acceptance Rate: 23% (25/108).

19. *Foreseer: Workload-aware Data Storage for MapReduce*
Jia Zou, Juwei Shi, Tongping Liu, Zhao Cao, WangLiu Chen.
Proceedings of 35th International Conference on Distributed Computing Systems (**ICDCS 2015**), poster.

20. *Predator: Predictive False Sharing Detection (Citation: 18)*
Tongping Liu, Chen Tian, Ziang Hu, Emery D. Berger.
Proceedings of the 19th ACM SIGPLAN Symposium on Principles and Practice of Parallel Programming (**PPoPP'14**). Acceptance Rate: 16% (28/179).

21. *Dthreads: Efficient Deterministic Multithreading (Citation: 233)*
Tongping Liu, Charlie Curtsinger, Emery D. Berger.
Proceedings of the 23rd ACM Symposium on Operating Systems Principles (**SOSP'11**). Acceptance rate:18% (28/153).

22. *Sheriff: Precise Detection and Automatic Mitigation of False Sharing (Citation: 63)*
    Tongping Liu, Emery D. Berger.
    Proceedings of the 26th Annual ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (**OOPSLA'11**).
    Acceptance rate: 37% (61/166).

23. *Grace: Safe Multithreaded Programming for C/C++ (Citation: 301)*
    Emery D. Berger, Ting Yang, Tongping Liu, Gene Novark.
    Proceedings of the 24th Annual ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications (**OOPSLA'09**).
    Acceptance rate: 17% (25/144).

24. *Redline: First Class Support for Interactivity in Commodity Operating Systems (Citation: 81)*
    Ting Yang, Tongping Liu, Emery D. Berger, Scott F. Kaplan, J.Eloit B.Moss.
    Proceedings of the 8th USENIX Symposium on Operating Systems Design and Implementation (**OSDI'08**).
    Acceptance rate: 13% (26/193).

25. *Dynamic Detection of Stack Overflow*
    Tongping Liu.
    Journal of Computer Science(China). No.9, 2007.

RESEARCH SUPPORT (TOTAL: $1,300,042, MY SHARE: $800,131)

1. NSF IUSE: *An Educational Tool for Teaching and Learning Concurrent Computer Programming Techniques*, **Lead PI**, IUSE 2215193, Total $ 299,970 . 07/01/202-06/30/2025

2. NSF CCF Proposal: *SPX: Pinpointing and Resolving Scalability Culprits Hidden in Different Components of the Whole System Stack*, **Lead PI**, CCF-1823004, Total $999,883, My Share: $499,992, 10/01/2018-9/30/2023

3. Mozilla Faculty Research Grant: *Guarder: Defending Heap Vulnerabilities with Flexible Guarantee and Better Performance*, **Sole PI**, Amount $51,073, 12/2017-unlimited.

4. Google Faculty Award: *Efficient, Effective, and Intelligent False Sharing Detection*, **Sole PI**, Amount $42,355, 08/2015-unlimited.

5. NSF CCF Proposal: *CRII: Evidence-Assisted Detection and Elimination of Memory Errors*, **Sole PI**, CCF-1566154, $206,731, 03/01/2016-02/28/2019

TEACHING RELATED EXPERIENCE

Operated Systems (graduate): Spring 2015, Fall 2015, Fall 2017, Fall 2018, Spring 2022.

Operated Systems (undergraduate): Fall 2016, Spring 2017, Spring 2018, Fall 2018, Spring 2021.

Kernel Programming: Spring 2017, Spring 2018.

Parallel and Distributed Software Systems: Spring 2016.

Performant and Reliable Software Systems: Fall 2019.

Systems Programming: Fall 2014, Fall 2020.

Research Methods: Spring 2019.

STUDENTS SUPERVISION
Current Students:
Jin Zhou

Hanmei Yang

Jiaxun Tang

Mingcan Xiang

Alumni:
Hongyu Liu (Ph.D., 08/2015–12/2019)

Sam Silvestro (Ph.D., 01/2016–08/2019)

Jinpeng Zhou (Master thesis, 08/2015–06/2017)

Corey Crosser (Master Projects, graduated in 05/2018)

Kyle Haley (B.S., 2018)

PATENTS

1. Classification of Different Types of Cache Misses. Tongping Liu, Jin Zhou, Steven Tang, Hanmei Yang. US Application No. 63/281,942.

2. A System and Method for Memory Allocation and Management In Non-Uniform Memory Access Architecture Computing Environment. Tongping Liu, Xin Zhao. US Application No. 63/200,062.

3. A Precise and Fully-Automatic On-Site Failure Diagnosis Method. Tongping Liu, Hongyu Liu, Sam Silvestro. Application 62/685,853.

4. Low-Overhead Detection Techniques for Synchronization Problems in Parallel and Concurrent Software. Tongping Liu, Mohammad Mejbah ul Alam, Abdullah Muzahid. US Patent App. 16/385,118.

5. Guarder: An Efficient Heap Allocator with Strongest And Tunable Security. Tongping Liu, Sam Silvestro, Hongyu Liu. US Patent App. 16/656,853.

6. FreeGuard: A Faster Secure Heap Allocator. Tongping Liu, Sam Silvestro, Hongyu Liu. Patent No. 10,901,828.

7. Defeating Deadlocks in Production Software. Tongping Liu, Jinpeng Zhou, Sam Silvestro, Hongyu Liu. US Patent 10,915,424.

8. System and Method for Detecting False Sharing. Tongping Liu, Chen Tian, Ziang Hu. US Patent 9,678,883.

9. System and Method for Predicting False Sharing. Chen Tian, Tongping Liu, Ziang Hu. US Patent 9,547,599.

10. Prevention of race conditions in library code through memory page-fault handling mechanisms. Daniel G. Waddington, Chen Tian, Tongping Liu. US Patent App. 13/425,312.

11. Coupled Lock Allocation and Lookup for Shared Data Synchronization in Symmetric Multithreading Environments. Daniel G. Waddington, Tongping Liu, Chen Tian. US Patent 8,868,849.

12. Mapping Guest Pages to Disk Blocks to Improve Virtual Machine Management Process. Kiran Tati, Rajesh Venkatasubramanian, Carl A. Waldspurger, Alexander Thomas Garthwaite, Tongping Liu. US Patent 10,474,369.

CONTRIBUTIONS ON OPEN-SOURCE-PROJECTS

1. FreeGuard: A Faster Secure Heap Allocator.
   https://github.com/UTSASRG/FreeGuard.

2. UnDead: Detecting and Tolerating Deadlocks in Production Software.
   https://github.com/UTSASRG/UnDead.

3. SyncPerf: Categorizing, Detecting, and Diagnosing Synchronization Performance Bugs.
   https://github.com/UTSASRG/SyncPerf.

4. DOUBLETAKE: Fast and Precise Error Detection via Evidence-Based Dynamic Analysis.
   https://github.com/plasma-umass/predator.

5. PREDATOR:Predictive False Sharing Detection: main contributor.
   https://github.com/plasma-umass/predator.

6. Sheriff: Precise Detection and Automatic Mitigation of False Sharing, main contributor.
   https://github.com/plasma-umass/sheriff.

7. DTHREADS project (A robust deterministic multithreading): main contributor.
   https://github.com/plasma-umass/dthreads.

8. SANPPLE project: one of main contributors.
   http://en.sourceforge.jp/projects/sfnet_snapple/releases/.

INVITED TALKS

1. *MemPerf: Identify Performance Slowdown Caused by Memory Allocators*. University of California, Merced, October 2021.

2. *Evidence-Based Error Detection and Diagnosis*. North Carolina State University, October 2020.

3. *In-Situ Failure Diagnosis*. Rice University, May 2019.

4. *Defending Memory Vulnerabilities Latent in Production Software*. Hong Kong University of Science and Technology, March 2019.

5. *Defending Memory Vulnerabilities Latent in Production Software*. Simon Fraser University, March 2019.

6. *Defending Memory Vulnerabilities Latent in Production Software*. University of California Merced, March 2019.

7. *Evidence-Based Error Detection and Diagnosis*. University of Toronto, March, 2019.

8. *Evidence-Based Error Detection and Diagnosis*. University of Massachusetts Amherst, February, 2019.

9. *Evidence-Based Error Detection and Diagnosis*. Stevens Institute of Technology, February, 2019.

10. *Evidence-Based Error Detection and Diagnosis*. University of Massachusetts Lowell, February, 2019.

11. *iReplayer: In-situ Replaying Errors of Multithreaded Programs*. Texas State University, September 2017.

12. *SyncPerf: Categorizing, Detecting, and Diagnosing Synchronization Performance Bugs*. Shenzhen Institutes of Advanced Technology, June 2017.

13. *SyncPerf: Categorizing, Detecting, and Diagnosing Synchronization Performance Bugs*. Xiamen University, June 2017.

14. *iReplayer: Identical and Efficient Record-And-Replay*. Harbin Institute of Technology, May 2017.

15. *SyncPerf: Categorizing, Detecting, and Diagnosing Synchronization Performance Bugs*. Harbin Engineering University, May 2017.

16. *iReplayer: Identical and Efficient Record-And-Replay*. George Mason University, October 2016.

17. *iReplayer: Identical and Efficient Record-And-Replay*. George Washington University, October 2016.

18. *Deterministic Systems to Defeat Reliability Problems*. South China University of Technology, December 2015.

19. *Deterministic Systems to Defeat Reliability Problems*. Guangdong University of Technology, December 2015.

20. *Deterministic Systems to Defeat Reliability Problems*. University of Science and Technology Beijing, December 2015.

21. *Improving the Performance of Parallel Applications*. Beijing University of Posts and Telecommunications, December 2015.

22. *Improving the Performance of Parallel Applications*. Institute of Computing Technology, December 2015.

23. *Improving the Performance of Parallel Applications*. Chinese Academy of Science (ICT), December 2015.

24. DOUBLETAKE: *Efficiently Detecting Memory Errors*. Huawei US R&D Center, February 2015.

25. *Performance Improvement for Parallel Applications*. Washington State University

26. *Performance Improvement for Parallel Applications*. HP Labs, April 2014.

27. *Performance Improvement for Parallel Applications*. NEC Labs America, March 2014.

28. *Performance Improvement for Parallel Applications*. North Carolina State University, March 2014.

29. *Performance Improvement for Parallel Applications*. College of William and Mary, March 2014.

30. *Performance Improvement for Parallel Applications*. University of Texas at San Antonio, February 2014.

31. *Performance Improvement for Parallel Applications*. Florida International University, February 2014.

32. PREDATOR: *Predictive False Sharing Detection*. Huawei US R&D Center, August 2013.

33. *Safe and Efficient Multithreading*. Huawei US R&D Center, June 2013.

34. DTHREADS: *Efficient Deterministic Multithreading*. IBM T.J.Watson Lab, August 2012.

35. SHERIFF: *Precise Detection and Automatic Mitigation of False Sharing*. IBM T.J.Watson Lab, August 2012.

36. SHERIFF: *Precise Detection and Automatic Mitigation of False Sharing*. University of Texas at Austin, January 2011.