SATISH NARAYANASAMY

Phone : (858) 401-0455
Fax : (734) 763-4617
nsatish@umich.edu
http://www.eecs.umich.edu/~nsatish

Office: BBB 4721
University of Michigan
2260 Hayward Street
Ann Arbor, MI 48109-2121

EDUCATION

PhD Computer Science — Dec. 2007
University of California, San Diego
Dissertation: "Deterministic Replay Using Processor Support and Its Applications"
Advisor: Professor Brad Calder

MS Computer Science — June 2005
University of California, San Diego

BE Computer Science and Engineering — April 2001
College of Engineering, Anna University, Chennai
Thesis: Decentralized Computing.

PROFESSIONAL EMPLOYMENT

University of Michigan, Ann Arbor — Sep. 2014 - Present
Associate Professor
Computer Engineering Lab Director
Electrical Engineering and Computer Science

Sequal Inc. — June 2017 - Present
President, co-founder

University of Michigan, Ann Arbor — Jan. 2008 – Aug. 2014
Morris Wellman Assistant Professor
Electrical Engineering and Computer Science

Microsoft Research, Bangalore — Aug. 2015 – Oct. 2015
Consultant

Microsoft Research, Redmond — Jul. 2013 – Aug. 2013
Visiting Researcher

Microsoft Research, Redmond — Sep. 2007 – Dec. 2007
Visiting Researcher

Microsoft, Redmond — Summer 2006
Intern, Center for Software Excellence

Intel, Santa Clara June 2003 - May 2004
Intern, Microprocessor Research Lab

HONORS AND AWARDS

Morris Wellman Faculty Development Assistant Professor 2014
Awarded to a junior faculty member in Computer Science and Engineering to recognize outstanding contributions to teaching and research.

Google Research Faculty Award 2014

IEEE Micro Top Picks Award 2013
"A Safety-First Approach to Memory Models"

NSF CAREER 2012
"Holistic System Solutions for Empowering Parallel Programmers"

Best Paper Award, ASPLOS 2011
"DoublePlay: Parallelizing Sequential Logging and Replay"

IEEE Micro Top Picks Award 2006
"Patching Processor Design Errors"

IEEE Micro Top Picks Award 2005
"BugNet: Continuously Recording Program Execution for Deterministic Replay Debugging"

IEEE Top Picks recognizes papers "most relevant to industry and significant in contribution to the field of computer architecture" every year.

Best Paper Nomination, DATE 2011
"Heterogeneous Main Memory"

STUDENT HONORS AND AWARDS

ProQuest Dissertation Award 2013
Dongyoon Lee, "Holistic System Design for Deterministic Replay".
University of Michigan awarded only ten dissertations, selected from over 650 dissertations across all disciplines in 2013. ProQuest Distinguished Dissertation Award recognizes exceptional and unusually interesting work produced by doctoral students in the last phase of their graduate work.

VMWare Fellowship 2012
Dongyoon Lee. VMWare awarded two fellowships to outstanding students pursuing research related to VMware's business interests.

Rackham International Student Fellowship 2009
Jie Yu. Rackham International Student Fellowship assists outstanding international students.

RESEARCH INTERESTS AND SIGNIFICANT CONTRIBUTIONS

I work at the intersection of computer architecture, software systems and compilers. My current interests include concurrent systems, specialized accelerator architectures and systems for mobile and web applications, Big Data for program analysis and MOOC tools, and system reliability.

Multi-processor Deterministic Replay: I developed several processor-assisted recording solutions for deterministically replaying a parallel program. Since the 1980s, processors have added little to no support for helping programmers improve software reliability. Given the gravity of the parallel programming problem that we were facing, it was high time to change that trend. I was part of a few researchers that started this discussion in the early part of the last decade, and it is gratifying to see industry and academia investing significant effort today to address this challenge.

At Michigan, colleagues and I drastically reduced the performance overhead of software multiprocessor replay solutions from over 100x to about ~2% for I/O intensive servers and ~2x for scientific applications.

Technology Transfer: In collaboration with Intel, we developed a software replayer named PinPlay using our0020BugNet's load-value logging solution. Intel used PinPlay for trace-based processor simulation across different platforms and debug commercial workloads. PinPlay integrated with our concurrency testing tool Maple has now been released to public.

BugNet's logging scheme also influenced the implementation of Microsoft's iDNA replay solution. I worked with Microsoft engineers to use iDNA's replayer to filter programmer intended data-races and accurately find high priority concurrency bugs. The tool that I helped develop was used to find and fix over 1000 bugs in Windows software.

Memory Models: Current Java and C++0x memory models provide weak or no semantics for racy programs, which complicates debugging and could compromise security. Our work disproved the conventional wisdom by showing that an SC-preserving compiler and hardware together costs only about 6% performance overhead. This result could eventually help us guarantee language-level SC.

Concurrency Testing Tools: Maple, a concurrency tool for testing untested thread interleavings, and a set of standard concurrency analysis tools (data race detection, systematic testing, probabilistic concurrency testing, etc.) was open-sourced. Also, we published many concurrency bugs that we discovered in our project through an online repository. This infrastructure has been used by researchers from several institutions including Illinois, Washington, Princeton, Microsoft, Google, Tsinghua, and Fudan, to build and evaluate concurrency testing tools.

Correctly Executing Unreliable Concurrent Software: Yu and I introduced the idea of thread interleaving memoization. We showed that by remembering tested thread schedules and avoiding untested schedules at runtime, we can tolerate a large fraction of concurrency errors. Colleagues and I also developed an operating system solution that exercised complementary thread schedules during production runs to detect concurrency bugs and avoided them by picking a schedule that is more likely to be correct.

Energy Proportional Multi-Core Systems: While some applications (e.g., playing a video) may be data-intensive and demand high bandwidth, several others may not (e.g., text editor). However, today's homogenous memory and interconnect designs are not energy proportional, as their power consumption does not reduce proportionately while executing applications with a lower data bandwidth demand. We proposed a heterogeneous main memory with mobile and server DRAMs, and a power-gating solution for the on-chip interconnect to help achieve energy proportionality.

Accelerators and Tools for Event-Driven Mobile and Web Applications

A wide range of computing systems built today, from mobile and web applications to Internet-of-Things to distributed servers (e.g., Twitter), use the asynchronous or event-driven model. Event-driven programming model naturally supports systems that receive asynchronous input from a rich array of sensors and user input modalities. In spite of its wide spread use, surprisingly, existing systems and debugging tools are optimized only for synchronous programs, and have ignored the unique characteristics of asynchronous programs.

This project aims to accelerate asynchronous programs by optimizing the computing stack and also develop a treasure chest of programmer productivity tools for them. As a first step, we have developed a causality model and a tool for detecting concurrency errors due to asynchrony for the Android event-driven applications [PLDI 2014]. We are also investigating customized runtime systems and processor architectures for improving the interactive performance of asynchronous programs such as web applications [PACT 2014].

Unified Abstractions for Heterogeneous Accelerators

As we customize the computing stack for different domains, it is imperative that we define unified abstractions that can enable diverse threads from these domains to co-exist and communicate with each other. A unified and coherent single address space for an application can enable programmers to easily transfer data between its heterogeneous threads. This would require new coherence protocols that are customized for the characteristics of varied accelerators. Concurrency memory model semantics need to be carefully defined to integrate legacy processor architectures (x86, ARM) with new accelerator interfaces. Operating systems should be able to efficiently provide functionalities such as virtual memory and thread pre-emptions on diverse processor architectures. Garbage collectors should be able to manage data stored in heterogeneous memory systems. This project is investigating vertically integrated solutions to address these problems.

BigCode and BigData for Program Analysis and MOOC Tools

We postulate that a corpus-driven ``big-code'' approach can work for program analysis. Today, one can easily gather billions of lines of program text from various sources such as online repositories, web applications, and MOOC courses. By examining these large corpora of source code and computing the probability of observing programming patterns, we can build a statistical body of knowledge and use it to improve a wide range of program analysis. For example, we may be able to support *auto-correct* and *auto-complete* features (popular in smartphone text editors) in program development environments. *Programming through speech* can become a reality. We can build scalable pedagogical tools such as auto-graders and plagiarism checkers [OOPSLA 2014] for massive open online courses (MOOCs).

As low-overhead replay systems mature, it may be possible to record all program executions on a system. Large data-centers already collect PetaBytes of logs every day from its executions. Software companies such as Microsoft and Google receive millions of trace and core dumps from its users every day. We are currently investigating BigData methods for analyzing this massive amount of execution data to find errors, security breaches, improve performance, and reduce burden on testers.

SOFTWARE AND OTHER ARTIFACTS

PinPlay Record and Replay System

I worked with Intel researchers (Drs. Patil, Pereira, and Cohn) to develop a software-based replay solution using the logging scheme that we had proposed in the BugNet architecture [Narayanasamy et al., ISCA'05, IEEE Micro Top Picks Award]. This tool was originally named as pinSEL [Narayanasamy et al., Sigmetrics'06]. It helped Intel support trace-based processor simulation of applications developed for different operating systems and do away with their tedious system emulation support in their architectural simulators such as ASIM. Intel continued to improve this tool to add functionalities such as support for debugging commercial workloads. Intel released PinPlay for public use.
URL: http://software.intel.com/en-us/articles/program-recordreplay-toolkit

Maple Concurrency Analysis Framework

An x86 dynamic analysis framework for concurrent programs built on top of Intel's Pin tool. It can be used to quickly build new dynamic analysis tools for concurrent programs. It also consists of several concurrency analysis tools (data race detection, systematic testing, probabilistic concurrency testing, etc.), including our coverage-driven testing tool [OOPSLA 2012]. Maple has now been integrated with PinPlay.
URL: https://github.com/jieyu/maple

Online Concurrency Bug Repository

This online repository contains a detailed description of several concurrency bugs discovered in production open-source server and client applications such as Apache, MySQL, Memcached, etc. It has been used by researchers from several institutions including Illinois, Washington, Princeton, Microsoft, Google, Tsinghua, Fudan, etc., to build and evaluate concurrency testing tools.
URL: https://github.com/jieyu/concurrency-bugs

SC-preserving LLVM compiler

Colleagues and I debunked the myth that preserving sequential consistency (SC) semantics during compiler optimizations could be prohibitively expensive. Our extension to the LLVM compiler that preserves SC has been released to the public. It has been used by researchers externally (Illinois) to conduct their memory model research.
URL: http://web.eecs.umich.edu/~nsatish/llvmsc.tar.gz

TEACHING

Significantly revised the course contents and revamped the projects for the EECS 483 course on "Compiler Construction''. This course was granted Major Design (MDE) status for winter 2011.

Introduced a new EECS 598 course on ``Ubiquitous Parallelism''. It discussed solutions that spanned across the layers of computing stack to address challenges in building concurrent systems.

Courses Taught

EECS 370 – Introduction to Computer Organization W'08, F'11, F'12, F'13, W'15, F'16
F 2013: Q1: 4.58; Q2: 4.76; Enrollment: 346

EECS 483 – Compiler Construction W 2009, W 2010, W 2011, W 2013
W 2013: Q1: 4.38; Q2: 4.40; Enrollment: 46

EECS 570 – Parallel Computer Architecture F 2009, W 2014, W'18
W 2014: Q1: 4.60; Q2: 4.73; Enrollment: 45

EECS 497 – Major Design Engineering F'17

EECS 598 – Special Topics: Ubiquitous Parallelism F 2010 and W 2012

Ubiquitous Parallelism, Beihang University Summer 2012

Q1: OVERALL, THIS WAS AN EXCELLENT COURSE (5: EXCELLENT; 4: VERY GOOD)
Q2: OVERALL, THE INSTRUCTOR WAS AN EXCELLENT TEACHER (5: EXCELLENT; 4: VERY GOOD)

RESEARCH ADVISING

Doctoral Students

Dongyoon Lee
"Hybrid Program Analysis for Determinism"
Graduated in 2013
ProQuest Dissertation Award
VMWare Fellowship
Assistant Professor, Virginia Tech.

Jie Yu
"Finding and Tolerating Concurrency Bugs"
Graduated: 2013
Awarded Rackham International Student Fellowship
Software Engineer, Twitter.

Abhayendra Singh
"Safety-First Approach to Memory Models"
Graduated in 2015
Google

Gaurav Chadha (co-advised by Scott Mahlke)
"Accelerators for Web Applications"
Graduate in 2015
Oracle

Chun-Hung Hsiao
"Taming Concurrency in Event-Driven Systems"
Candidate; Advisee since 09/2010; Expected Graduation: 2017

Shaizeen Aga
"Near Memory Computing for Big Data Applications"
Pre-Candidate; Advisee since 09/2011; Expected Graduation: 2017

Hanyun Tao
Advisee since 06/2015; Expected Graduation: 2020

Hossein Golestani
Advisee since 09/2016; Expected Graduation: 2021

Subarno Banerjee
Advisee since 08/2016; Expected Graduation: 2021

Visiting Scholar

Lan Gao
"Coherence for Heterogeneous Processors"
Currently enrolled for PhD at Beihang University.
Visited from 09/2014 to present

Bo Liu
"Symbolic Analysis for Isolating Root Causes of Concurrency Bugs"
Currently enrolled for PhD at Shanghai Jiao Tong University.
Visited from 09/2010 to 08/2012

Masters Students

Ashutosh Parkhi
"Efficient Trace Compilation for Javascript Applications"
Graduated 04/2012

Felix Loh
"Heterogeneous Memory"
Graduated 06/2009

Pankit Thapar
"Active Scheduling for Multithreaded Program Testing"
Directed study, fall 2012

Jay Bhukhanwala
"Active Scheduling for Multithreaded Program Testing"

Directed study, fall 2012

Undergraduate Students

Allen Chang
"Quantifying Crash Latency of Software Bugs", summer 2008

Yuanbo Fan
"Sampling based data race detector using Phoenix Compiler", summer 2009

Haixuan Sun
"Sampling based data race detector using Phoenix Compiler", summer 2009

Yuandong Zhuang
"Vectorizing Medical Imaging Application on Windows Platform", summer 2009

Yan Zhang
"Heterogeneous Memory", directed study in fall 2011, summer 2012

Noam Samuel
"Hybrid Program Analysis for Data-Race Detection", winter 2012

Ziyun Kong
"Classifying Intended Races in Intel Inspector", summer 2012

Hanyun Tao
"Exploiting Event-Level Parallelism in Asynchronous Programs", summer 2014

Doctoral Committees

Hyunchul Park,	U of Michigan
Mojtaba Mehrara,	U of Michigan
Xi Chen,	U of Michigan
Mona Attariyan,	U of Michigan
Benjamin Wester,	U of Michigan
Ankit Sethia,	U of Michigan
Hyoun Kyu Cho,	U of Michigan
Aasheesh Kolli,	U of Michigan
Xuehai Qian,	UIUC
Nima Honarmand,	UIUC

RESEARCH GRANTS AND GIFTS

National Science Foundation (NSF), CCF
"SHF: Medium: Optimistic Static Analysis"
PI: Satish Narayanasamy
$1,079,989; 2017-present

Center for Future Architecture Research (a DARPA and MARCO funded center)
PI: Satish Narayanasamy
$300,000; 2015-2017

National Science Foundation (NSF), CCF
"SHF: Small: Accelerating Asynchronous Programs through Synergistic Hardware/Software Customization"
PI: Satish Narayanasamy
$450,000; 2015-present

Google Research Faculty Award
PI: Satish Narayanasamy
$75,000; 2014-present

Intel Corporation
"Taming Concurrency in Event-Driven Systems"
PI: Satish Narayanasamy
$70,000; 2014-present

National Science Foundation (NSF), CCF
"CAREER: Holistic System Solutions for Empowering Parallel Programmers"
PI: Satish Narayanasamy
$538,671; 2012-17.

National Science Foundation (NSF), CCF
"SHF: Small: Interleaving Constrained Parallel Runtime System for Tolerating Concurrency Bugs"
PI: Satish Narayanasamy
$499,946; 2009-12.

National Science Foundation (NSF), CNS
"CSR: Medium: Improving Software Reliability and Security through Multicore Technology"
PI: Peter Chen, Co-PIs: Jason Flinn and Satish Narayanasamy
$1,200,000; 2009-13.

Intel Corporation
"Active Testing Based on Interleaving Idioms for Parallel Programs"
PI: Satish Narayanasamy
$70,000; 2010-present

Intel Corporation
"Active Testing Based on Interleaving Idioms for Parallel Programs"
PI: Satish Narayanasamy

$70,000; 2011-present

Microsoft, Gift,
PI: Satish Narayanasamy
$10,000, 2008

Intel Corporation
"Efficient Execution of Medical Imaging Applications on the Intel Larrabee System"
PI: Satish Narayanasamy, Co-PIs: Jeff Fessler and Scott Mahlke
$135,000 plus equipment, 2008-11.

Microsoft, Gift,
PI: Satish Narayanasamy
$20,000, 2008

Intel, Equipment Grant
PI: Satish Narayanasamy
$20,000, 2008

Intel Corporation
"Optimizing Hadoop Applications Using Message-Passing Support in Intel's SCC"
PI: Satish Narayanasamy, Co-PI: Michael Cafarella
Access to 48-core Single-Chip Cloud Computer (SCC) cluster; 2010.

PROFESSIONAL SERVICE

University Service

Faculty Search Committee,	2017-present, 2013-14
MIDAS FERC,	2017-present
CE lab director	2017-present
Masters Advisor,	2014-2017
Undergraduate Advisor, Computer Engineering,	2010-12, 2013-14
Graduate Admissions Committee,	2008-10, 2012-13

Conference and Workshop Organization

Workshop and Tutorial Chair
International Symposium on High Performance Computer Architecture (HPCA) 2014

Program Co-chair
11th International Workshop on Dynamic Analysis (WODA) 2013
Co-located with ASPLOS 2013.

Publications Chair
International Symposium on Principles and Practice of Parallel Programming (PPoPP) 2010

Associate Editor

Computer Architecture Letters (CAL) 2017-present

Member of Program Committee

2018

Intl. Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)

2017

Symposium on Principles and Practice of Parallel Programming (PPoPP)

2016
Intl. Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)
Top Picks from the Computer Architecture Conferences (IEEE MICRO Top Picks)
International Symposium on Performance Analysis of Systems and Software (ISPASS)

2015
International Symposium on High Performance Computer Architecture (HPCA)

2014
Intl. Conference on Arch. Support for Prog. Languages and Operating Systems (ASPLOS)
Top Picks from the Computer Architecture Conferences (IEEE MICRO Top Picks)
Intl. Symposium on Code Generation and Optimization (CGO)
International Conference on Parallel Processing (ICPP)
Intl. Conference on High-Performance and Embedded Arch. and Compilers (HiPEAC)

2013
Programming Language Design and Implementation (PLDI)
International Symposium on Microarchitecture (MICRO)
International Symposium on High Performance Computer Architecture (HPCA)
Intl. Symposium on Code Generation and Optimization (CGO)
Intl. Conference on High-Performance and Embedded Arch. and Compilers (HiPEAC)
Workshop on Determinism and Correctness in Parallel Programming
Workshop on Memory Systems Performance and Correctness (MSPC)

2012
Intl. Symposium on Computer Architecture (ISCA)
International Symposium on High Performance Computer Architecture (HPCA)
Intl. Conference on Parallel Architectures and Compilation Techniques (PACT)
Intl. Conference on High-Performance and Embedded Arch. and Compilers (HiPEAC)
Intl. Conference on Computer Design (ICCD)
Workshop on Memory Systems Performance and Correctness (MSPC)

2011
Intl. Conference on Parallel Architectures and Compilation Techniques (PACT)

Intl. Symposium on Performance Analysis of Systems and Software (ISPASS)
Intl. Conference on Computer Design (ICCD)
Intl. Conference of VLSI Design
Network and Parallel Computing (NPC)
Workshop on Memory Systems Performance and Correctness (MSPC)
Workshop on Parallel Execution of Sequential Programs on Multi-core Arch. (PESPMA)

2010
Intl. Symposium on Code Generation and Optimization (CGO)
Intl. Conference on Computer Design (ICCD)
Intl. Conference of VLSI Design
Intl. Congress on Computer Applications and Computational Science (CACS)

2009
Intl. Conference on Computer Design (ICCD)
Workshop on Compiler and Arch. Tech. for Application Reliability and Security (CATARS)

Member of External Review Committee (ERC)

International Symposium on Microarchitecture (MICRO), 2017
Intl. Symposium on Computer Architecture (ISCA), 2017
International Symposium on Microarchitecture (MICRO), 2015
Intl. Symposium on Computer Architecture (ISCA), 2015
Programming Language Design and Implementation (PLDI), 2015
USENIX Symposium on Operating Systems Design and Implementation (OSDI), 2014
Intl. Conference on Parallel Architectures and Compilation Techniques (PACT), 2013
Intl. Conf. on Arch. Support for Prog. Languages and Operating Systems (ASPLOS), 2013
Programming Language Design and Implementation (PLDI), 2012
International Symposium on Microarchitecture (MICRO), 2012
Intl. Conf. on Arch. Support for Prog. Languages and Operating Systems (ASPLOS), 2011

Funding Panels and Grant Reviews

Panelist, CISE/CCF, National Science Foundation, 2017
Panelist, CISE/CCF, National Science Foundation, 2016
Panelist, CISE/CCF, National Science Foundation, 2016
Panelist, CISE/CCF, National Science Foundation, 2014
Panelist, CISE/CCF, National Science Foundation, 2013
Panelist, CISE/CCF, National Science Foundation, 2012
Panelist, CISE/CNS, National Science Foundation, 2011
Reviewer, U.S.-Israel Binational Science Foundation, 2011
Panelist, CISE/CCF, National Science Foundation, 2010
Panelist, CISE/CNS, National Science Foundation, 2010
Panelist, CISE/CCF, National Science Foundation, 2009

Reviewer

Journals: IEEE Micro Top Picks, TACO, TODAES, SPE, IEEE TPDS, IEEE Transaction on Computers, ACM TOSEM, ISRN Software Engineering

External reviewer for many conferences

PRESENTATIONS AND INVITED LECTURES

Invited Delegate

"Failure is not an option: Popular Parallel Programming",
CRA-CCC Workshop on Advancing Computer Architecture Research (ACAR-1),
San Diego, February 2010.

Invited Talks

"Taming Concurrency"

Cornell, 2015
U. Rochester, 2015
UCLA, 2015
Georgia Tech., 2015
John Hopkins, 2015

"Efficient Race Detection for Event-Driven Applications"

Intel, February 2015

"Call to arms: Customizing System Stack for Event-Driven Programs"

Center for Future Architecture Research (CFAR) annual meeting, November 2014

"Programmer Productivity Tools for Mobile and Web Applications"

Intel, January 2014

"System Design for Event-Driven Programs"

Intel, January 2014
Mining and Understanding Software Enclaves (MUSE), DARPA, March 2014

"Taming Concurrency"

Boston University, 2013

"Path to Achieving Sequential Correctness and Parallel Performance"

École polytechnique fédérale de Lausanne (EPFL), October 2013
6th International Workshop on Exploiting Concurrency Efficiently and Correctly, (EC)^2 held in conjunction with CAV 2013
Chinese Academy of Sciences, 2012
Beihang University, 2012

University of Pennsylvania, Nov 2011
Duke University, Durham, March 2011
NEC Research Labs, Princeton, April 2011

"Deterministic Replay and Uniparallelism"

Microsoft Research, Redmond, February 2011

"A Case against Unbridled Parallelism"

Qualcomm, Santa Clara, January 2011
Intel, Santa Clara, February 2010
University of Texas, Austin, October 2009
University of Wisconsin, Madison, September 2009

"Interleaving Idiom-Driven Active Testing for Multithreaded Programs"

Intel, August 2010

"Solving Software Reliability Issues Using Processor Support"

Indian Institute of Technology (IIT), Chennai, Aug 2008
Anna University, Chennai, Aug 2008
Intel, Oregon, June 2008
Intel, Boston, February 2008

"Looking Beyond Performance: Processors for Time Travel"

Intel Research, Pittsburgh, June 2007
University of Virginia, Charlottesville, June 2007
Columbia University, New York, April 2007.
Rice University, Houston, April 2007.
Texas A&M University, College Station, April 2007.
University of Michigan, Ann Arbor, March 2007.
University of Illinois, Urbana-Champaign, March 2007
Microsoft Research, Redmond, March 2007.
Washington University, Saint Louis, March 2007.
Ohio State University, Columbus, March 2007.
North Carolina State University, Raleigh, March 2007.
Intel, Oregon, Feb 2007.

"Catching Accurate Profiles in Hardware"

Intel, Santa Clara. March 2004

Conference Talks

"A Case for an Interleaving Constrained Shared-Memory Multi-Processor"
ISCA, Austin, 2009

"Automatically Classifying Benign and Harmful Data Races Using Replay Analysis"

PLDI, San Diego, 2007

"Patching Processor Design Errors"
ICCD, San Jose, 2006

"Recording Shared Memory Dependencies Using Strata"
ASPLOS, San Jose, 2006

"BugNet: Continuously Recording Program Execution for Deterministic Replay Debugging"
ISCA, Madison, 2005

"A Dependency Chain Clustered Microarchitecture"
IPDPS, Denver, 2005

"Creating Converged Trace Schedules Using String Matching"
HPCA, Madrid, 2004

"Catching Accurate Profiles in Hardware"
HPCA, Anaheim, 2003

PUBLICATIONS

Names of my graduate student advisees are underlined.

Journals

[J1] Daniel Marino, Abhayendra Singh, Todd D. Millstein, Madanlal Musuvathi, Satish Narayanasamy, "DRFx: An Understandable, High Performance, and Flexible Memory Model for Concurrent Languages". ACM Trans. Program. Lang. Syst. 38(4): 16:1-16:40 (2016).

[J2] Abhayendra Singh, Satish Narayanasamy, Dan Marino, Todd Millstein, and Madan Musuvathi, "A Safety-First Approach to Memory Models", IEEE Micro Special Issue: Top Picks from Computer Architecture Conferences, vol. 33, issue 3, pp. 96 – 104, May/June 2013.

[J3] Kaushik Veeraraghavan, Dongyoon Lee, Benjamin Wester, Jessica Ouyang, Peter M. Chen, Jason Flinn, Satish Narayanasamy, "DoublePlay: Parallelizing Sequential Logging and Replay", *ACM Transactions on Computer Systems (TOCS)*. vol. 30, no. 1, 2012.

[J4] Weihaw Chuang, Satish Narayanasamy, and Brad Calder, "Accelerating Meta Data Checks for Software Correctness and Security", *Journal of Instruction-Level Parallelism*, vol. 9, June 2007.

[J5] Smruthi Sarangi, Satish Narayanasamy, Bruce Carneal, Abhishek Tiwari, Brad Calder, and Josep Torrellas, "Patching Processor Design Errors Using Programmable Hardware", *IEEE Micro Special Issue: Top Picks from Computer Architecture Conferences*, vol. 27, no. 1, pp. 12-25, January/February 2007.

[J6] Satish Narayanasamy, Gilles Pokam, and Brad Calder, "BugNet: Recording Application Level Execution for Deterministic Replay Debugging", *IEEE Micro Special Issue: Top Picks from Computer Architecture Conferences*, vol. 26, no. 1, pp. 100-109, January/February 2006.

Refereed Conferences

[C1] David Devecsery, Peter Chen, Satish Narayanasamy, Jason Flinn, "Fast, Sound Dynamic Analysis through Predicated Static Analysis and Speculative Execution". Conditionally accepted to appear in *ACM International Conference on Architectural Support for Programming Languages and Operating Systems (**ASPLOS**)*, March 2018.

[C2] Shaizeen Aga and Satish Narayanasamy. "InvisiMem: Smart Memory Defenses for Memory Bus Side Channel". In *International Symposium on Computer Architecture (**ISCA**)*, June 2017.

[C3] Aasheesh Kolli, Vaibhav Gogte, Ali Saidi, Stephan Diestelhorst, Peter M. Chen, Satish Narayanasamy, Thomas F. Wenisch. In *International Symposium on Computer Architecture (**ISCA**)*, June 2017.

[C4] Chun-Hung Hsiao, Satish Narayanasamy, Essam Khan, Cristiano Pereira, and Gilles Pokam, "AsyncClock: Scalable Inference of Asynchronous Event Causality", In *22nd ACM International Conference on Architectural Support for Programming Languages and Operating Systems (**ASPLOS**)*, April 2017.

[C5] Shaizeen Aga, Supreet Jeloka, Arun Subramaniyan, Satish Narayanasamy, David Blaauw, and Reetuparna Das. "Compute Caches", In *23rd IEEE Symposium on High Performance Computer Architecture (**HPCA'17**)*, February 2017.

[C6] Shaizeen Aga, Sriram Krishnamoorthy, Satish Narayanasamy, "CilkSpec: optimistic concurrency for Cilk", In *Proceedings of the International Conference for High Performance Computing, Networking, Storage and Analysis* (***SC),*** November 2015.

[C7] Abhayendra Singh, Shaizeen Aga, Satish Narayanasamy, "Efficiently enforcing strong memory ordering in GPUs", In *International Conference on Microarchitecture (**MICRO**)*, October 2015.

[C8] Gaurav Chadha, Scott Mahlke, Satish Narayanasamy, "Accelerating Asynchronous Programs through Event Sneak Peak", In *International Symposium on Computer Architecture (**ISCA**)*, June 2015.

[C9] Shaizeen Aga, Abhayendra Singh, Satish Narayanasamy, "zFence: Data-less Coherence for Efficient Fences", In *International Conference on Supercomputing (**ICS**)*, June 2015.

[C10] Dan Marino, Todd Millstein, and Madan Musuvathi, Satish Narayanasamy, Abhayendra Singh, "The Silently Shifting Semicolon", In *Inaugural Summit on Advances in Programming Languages (**SNAPL**)*, May 2015.

[C11] Chun-Hung Hsiao, Michael Cafarella, Satish Narayanasamy, "Using Web Corpus Statistics for Program Analysis", In *ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages and Applications (**OOSPLA**)*, October 2014.

[C12] Chun-Hung Hsiao, Michael Cafarella, Satish Narayanasamy, "Reducing MapReduce Abstraction Costs for Text-Centric Applications", In *International Conference on Parallel Processing (**ICPP**)*, Sep 2014.

[C13] Gaurav Chadha, Scott Mahlke, Satish Narayanasamy, "EFetch: Optimizing Instruction Fetch for Event-Driven Web Applications", In *International Conference on Parallel Architectures and Compilation Techniques (**PACT**)*, August 2014.

[C14] Chun-Hung Hsiao, Jie Yu, Satish Narayanasamy, Ziyun Kong, Cristiano L. Pereira, Gilles A. Pokam, Peter M. Chen, Jason Flinn, "Race Detection for Event-Driven Mobile Applications", In Proceedings of the *ACM SIGPLAN Conference on Programming Language Design and Implementation (**PLDI**)*, June 2014.

[C15] Reetuparna Das, Satish Narayanasamy, Sudhir Satapathy, and Ron Dreslinski, "Energy Proportional Multiple Network-on-Chip", In *International Symposium on Computer Architecture (**ISCA**)*. June 2013.

[C16] Jessica Ouyang, Peter M. Chen, Jason Flinn, Satish Narayanasamy, "...and region serialiability for all", In *Proceedings of the USENIX Workshop on Hot Topics in Parallelism (**HotPar**)*, June 2013.

[C17] Benjamin Wester, David Devecsery, Peter M. Chen, Jason Flinn, and Satish Narayanasamy, "Parallelizing Data Race Detection", In *International Conference on Architectural Support for Programming Languages and Operating Systems (**ASPLOS**)*, March 2013.

[C18] Jie Yu, Satish Narayanasamy, Cristiano Pereira, and Gilles Pokam, "Maple: A coverage-driven testing tool for multithreaded programs", In *Proceedings of the Conference on Object-Oriented Programming Systems, Languages, and Applications (**OOPSLA**)*, October 2012.

[C19] Gaurav Chadha, Scott Mahlke, and Satish Narayanasamy, "When less is more (LIMO): Controlled parallelism for improved efficiency", In *Proceedings of the 2012 ACM International Conference on Compilers, Architectures and Synthesis for Embedded Systems* (***CASES***), October 2012.

[C20] Abhayendra Singh, Satish Narayanasamy, Dan Marino, Todd Millstein, and Madan Musuvathi, "End-to-end Sequential Consistency", In *Proc. of the 39th International Symposium on Computer Architecture (**ISCA**)*, June 2012. **IEEE Micro Top Picks Award**.

[C21] Dongyoon Lee, Peter M. Chen, Jason Flinn, and Satish Narayanasamy, "Chimera: Hybrid Program Analysis for Determinism", In *Proc. of the 34th ACM SIGPLAN Conference on Programming Language Design and Implementation (**PLDI**)*, June 2012.

[C22] Kaushik Veeraraghavan, Peter M. Chen, Jason Flinn, and Satish Narayanasamy, "Surviving and detecting data races using complementary schedules", In *Proc. Of the 23rd ACM Symposium on Operating Systems Principles (**SOSP**)*, October 2011.

[C23] Dan Marino, Abhayendra Singh, Todd Millstein, Madan Musuvathi, and Satish Narayanasamy, "A Case for SC-Preserving Compiler", In *Proc. of the 33nd ACM SIGPLAN Conference on Programming Language Design and Implementation (**PLDI**)*, June 2011.

[C24] Kaushik Veeraraghavan, Dongyoon Lee, Benjamin Wester, Peter M. Chen, Jason Flinn, and Satish Narayanasamy, "DoublePlay: Parallelizing sequential logging and replay", In *Proc. of the Sixteenth International Conference on Architectural Support for Programming Languages and Operating Systems (**ASPLOS**)*, March 2011. **Best Paper Award.**

[C25] Abhayendra Singh, Dan Marino, Satish Narayanasamy, Todd Millstein, and Madan Musuvathi, "Efficient Processor Support for a Memory Model with Exceptions", In *Proc. of the Sixteenth International Conference on Architectural Support for Programming Languages and Operating Systems (**ASPLOS**)*, March 2011.

[C26] Sujay Phadke and Satish Narayanasamy, "MLP-aware Heterogeneous Main Memory", In *Proc. of the Design, Automation & Test in Europe (**DATE**)*, March 2011. (**Best Paper Award Nomination** –5 out of 789 submissions).

[C27] Dongyoon Lee, Mahmoud Said, Satish Narayanasamy, and Zijiang (James) Yang, "Symbolic Analysis for Replaying under Total Store Order", In *Proc. of the 17th IEEE International Symposium on High Performance Computer Architecture (**HPCA**)*, February 2011.

[C28] Jie Yu and Satish Narayanasamy, "Tolerating Concurrency Bugs Using Transactions as Lifeguards", In *Proc. of the 42nd International Symposium on Microarchitecture (**MICRO**)*, December 2010.

[C29] Dan Marino, Abhayendra Singh, Todd Millstein, Madan Musuvathi, and Satish Narayanasamy, "DRFx: A Simple and Efficient Memory Model for Concurrent Programming Languages", In *Proc. of the 32nd ACM SIGPLAN Conference on Programming Language Design and Implementation (**PLDI**)*, June 2010.

[C30] Dongyoon Lee, Benjamin Wester, Kaushik Veeraraghavan, Satish Narayanasamy, Peter M. Chen, and Jason Flinn, "Respec: Efficient Online Multiprocessor Replay via Speculation and External Determinism", To appear in *Proc. of the Fifteenth International Conference on Architectural Support for Programming Languages and Operating Systems (**ASPLOS**)*, March 2010.

[C31] Dongyoon Lee, Mahmoud Said, Satish Narayanasamy, Zijiang (James) Yang, and Cristiano Pereira, "Offline Symbolic Analysis for Multi-Processor Replay", in *Proc. of the 41st International Symposium on Microarchitecture (**MICRO**)*, June 2009.

[C32] Jie Yu and Satish Narayanasamy, "A Case for an Interleaving Constrained Shared-Memory Multi-Processor", in *Proc. of the 36th International Symposium on Computer Architecture (**ISCA**)*, June 2009.

[C33] Daniel Marino, Madan Musuvathi, and Satish Narayanasamy, "LiteRace: Effective Sampling for Lightweight Data-Race Detection", In *Proc. of the 31st ACM SIGPLAN Conference on Programming Language Design and Implementation (**PLDI**)*, June 2009.

[C34] Satish Narayanasamy, Zhenghao Wang, Jordan Tigani, Andrew Edwards, and Brad Calder, "Automatically Classifying Benign and Harmful Data Races Using Replay Analysis", In *Proc. of the 29th ACM SIGPLAN Conference on Programming Language Design and Implementation (**PLDI**)*, June 2007.

[C35] Satish Narayanasamy, Ayse Coskun, and Brad Calder, "Predicting Faults Based on Anomalies in Speculative Execution", In *Proc. of Design, Automation, and Test in Europe (**DATE**)*, April 2007.

[C36] Weihaw Chuang, Satish Narayanasamy, Brad Calder, and Ranjit Jhala, "Bounds Checking with Taint-Based Analysis", In *Proc. of the International Conference on High Performance Embedded Architectures and Compilers (**HiPEAC**)*, January 2007.

[C37] Satish Narayanasamy, Cristiano Pereira, and Brad Calder, "Recording Shared Memory Dependencies Using Strata", In *Proc. Of the Twelfth International Conference on Architectural Support for Programming Languages and Operating Systems (**ASPLOS**)*, October 2006.

[C38] Weihaw Chuang, Satish Narayanasamy, Ganesh Venkatesh, Jack Sampson, Michael Van Biesbrouck, Gilles Pokam, Osvaldo Colavin, and Brad Calder, "Unbounded Page-Based Transactional Memory",

In *Proc. Of the Twelfth International Conference on Architectural Support for Programming Languages and Operating Systems (**ASPLOS**)*, October 2006.

[C39] Satish Narayanasamy, Bruce Carneal, and Brad Calder, "Patching Processor Design Errors", In *Proc. of the 24th International Conference on Computer Design (**ICCD**)*, October 2006. **IEEE Micro Top Picks Award**.

[C40] Satish Narayanasamy, Cristiano Pereira, and Brad Calder, "Software Profiling for Deterministic Replay Debugging of User Code", In *Proc. of the 5th International Conference on Software Methodologies, Tools and Techniques (**SoMET**)*, October 2006.

[C41] Satish Narayanasamy, Cristiano Pereira, Harish Patil, Robert Cohn, and Brad Calder, "Automatic Logging of Operating System Effects to Guide Application-Level Architecture Simulation", In *Proceedings of the Joint International Conference on Measurement and Modeling of Computer Systems, **SIGMETRICS**/Performance*, June 2006.

[C42] Satish Narayanasamy, Gilles Pokam, and Brad Calder, "BugNet: Continuously Recording Program Execution for Deterministic Replay Debugging", In *Proc. of the 32nd International Symposium on Computer Architecture (**ISCA**)*, June 2005. **IEEE Micro Top Picks Award**.

[C43] Satish Narayanasamy, Hong Wang, Perry Wang, John Shen, and Brad Calder, "A Dependency Chain Clustered Microarchitecture", In *Proc. of the 19th IEEE International Parallel and Distributed Processing Symposium (**IPDPS**)*, April 2005.

[C44] Satish Narayanasamy, Yuanfang Hu, Suleyman Sair, and Brad Calder, "Creating Converged Trace Schedules Using String Matching", In *Proc. of the 10th International Symposium on High Performance Computer Architecture (**HPCA**)*, February 2004.

[C45] Satish Narayanasamy, Timothy Sherwood, Suleyman Sair, Brad Calder, and George Varghese, "Catching Accurate Profiles in Hardware", In Proc. of the 9th International Symposium on High Performance Computer Architecture (HPCA), February 2003.

Others

[O1] Josep Torrellas, Mark Oskin, et al., "Failure is not an Option: Popular Parallel Programming", Report from *Computing Community Consortium (CCC) Workshop on Advancing Computer Architecture Research (ACAR-1)*, released October 2010.

[O2] Jessica Ouyang, Kaushik Veeraraghavan, Dongyoon Lee, Peter M. Chen, Jason Flinn, Satish Narayanasamy, "Epoch parallelism: One execution is not enough", In *Research Vision Session at 9th USENIX Symposium on Operating Systems Design and Implementation (**OSDI**)*, Oct 2010.

Patents

[P1] Shaizeen Aga, Abhayendra Singh, Satish Narayanasamy, "Efficient Memory Consistency Models Using Data-less Coherence", 2014.

[P2] Andrew J. Edwards et al., "Machine Instruction Level Race Condition Detection", *US Patent 7861118*, Published: December 2010.

[P3] Madan Musuvathi, Daniel Marino, and Satish Narayanasamy, “Sampling Techniques for Dynamic Data-Race Detection”, *US Patent Application 20100131931*, Published: May 2010.

[P4] Satish Narayanasamy et al., “System, Method and Apparatus for Dependency Chain Processing”, *US Patent 7603546*, Published: October 2009.

PERSONAL

Indian Citizen, Permanent Resident of the United States