

Prathmesh Kallurkar

Department of Computer Science and Engineering,
Indian Institute of Technology Delhi,
Hauz Khas, New Delhi - 110016, India.

Phone: (+91) 9891257591
prathmesh.kallurkar@cse.iitd.ac.in
www.cse.iitd.ac.in/~prathmesh

PERSONAL PROFILE

I am currently working as a CPU Research Scientist in the Microarchitecture Research Lab at Intel Labs, India. My research interests include computer architecture, systems software development and distributed systems.

EDUCATION

2012-Present	Ph.D. in Computer Science and Engineering Indian Institute of Technology Delhi. Thesis: Architectural Support for Enhanced Performance of OS Intensive Applications Advisor: Dr. Smruti R. Sarangi	GPA: 10/10
2010-2012	M.Tech in Computer Science and Engineering Indian Institute of Technology Delhi. Thesis: Design Space Exploration of OS Interference Aware Cache Advisor: Dr. Smruti R. Sarangi	GPA: 8.34/10
2006-2010	B.E. in Computer Science and Engineering Birla Vishvakarma Mahavidyalaya	GPA: 8.31/10

RESEARCH

COMPARING OPEN SOURCE OPERATING SYSTEMS	In this work, we perform an in-depth analysis of the execution of three popular open-source operating systems: Linux, FreeBSD, and OpenBSD for a suite of 7 server-class workloads. We use a full system simulation framework (Tejas) to analyze the execution of the entire system (application+OS) from an architectural perspective. This work seeks to bring out insights that can serve as directives to OS designers. For example, we show that owing to smaller i-cache and d-cache footprints (measured as number of unique cache lines accessed) of filesystem related system call and interrupt handlers, FreeBSD outperforms Linux and OpenBSD for file intensive workloads. We also compare two versions of these operating systems, one released in March-2014 and another released in March-2016 to analyze the net impact of all the OS modifications on the execution of selected workloads. <i>This is an ongoing work. We will submit it to the ACM Operating Systems Review (OSR) journal in the next 2-3 months.</i>
SCHEDULER FOR OS INTENSIVE APPLICATIONS	Traditional OS schedulers typically execute all types of tasks — user applications, system call and interrupt handlers — on the same core, thereby leading to i-cache pollution. This reduces the performance of OS intensive applications such as web servers and database servers by up to 50%. We propose <i>SchedTask</i> , a hardware-assisted OS scheduler that executes dissimilar tasks on separate cores. The primary contributions of the scheduler include a novel approach of using hardware Bloom filters to quantify the instruction similarity between tasks at run-time, and a work stealing algorithm that increases instruction throughput by scheduling suitable tasks on idle cores.

For a suite of 8 OS intensive applications, we show that SchedTask outperforms the state of the art OS schedulers by up to 29% (mean: 11.4%). *This work has been accepted for publication at the IEEE/ACM International Symposium on Microarchitecture (MICRO), 2017.*

INSTRUCTION PREFETCHER FOR OS INTENSIVE APPLICATIONS Computer architects have long used instruction prefetching to improve the performance of operating system (OS) intensive workloads. Sophisticated instruction prefetchers are implemented mostly in hardware; they record the execution history of a program in dedicated structures and use this information for prefetching if a known execution pattern is repeated; the storage overheads of these structures are prohibitively high (64- 200 KB per core). We show that in the case of OS intensive applications, the i-cache misses are mostly clustered in small execution blocks that follow OS events such as interrupts, system calls, and context switches. We propose a sophisticated technique to identify and prefetch these execution blocks using a combination hardware and software modifications. Our technique uses only 4 additional registers per core, and still gives a performance improvement of up to 14% (mean: 7%) over the state of the art instruction prefetchers for a suite of 8 OS intensive applications. *This work was accepted for publication by the IEEE/ACM International Symposium on Microarchitecture (MICRO), 2016.*

ETHICAL HACKING OF LICENSE MANAGERS This work proposes a novel strategy to circumvent the license validation of a proprietary software. We first collect the execution trace of a licensed software using an emulator such as Intel PIN or QEMU. These traces are then analyzed to identify regions of code that are responsible for the license checking. Such regions of code are then skipped in the next execution in a way that the software's data state remains valid, and it continues along the ideal execution path. As a proof of concept, we use these techniques to crack five applications protected using license managers. *This work was awarded the best poster award at the Security and Privacy Symposium (SPS), 2015.*

OS JITTER MITIGATION For a suite of 17 multi-threaded applications, we show that the OS activity (system call and interrupt handlers, and kernel threads) can cause timing variations of up to 35.22% in large multi-core systems. This timing variation is defined as Jitter. We propose to reduce the OS-induced jitter by using two hardware units: (1) a Jitter unit that uses a distributed protocol to keep track of the time lost to the OS and subsequently tries to compensate it by using DVFS, and (2) a dedicated cache to store the OS cache lines at the L2 level in memory hierarchy. The area overhead of our scheme is limited to 1% and is shown to reduce the overall jitter of a suite of 17 multi-threaded applications by an average 8%. *This work has been published in the IEEE Transactions on Parallel and Distributed Systems (TPDS) journal in 2014.*

PROCESSOR FOR CLOUDS This work proposes the design of TriKon, a heterogeneous processor for clouds. The primary contribution of the work is a novel cache called Triangle cache that stores the instructions of all codes in the software stack — application, operating system, and virtual machine monitor — in separate memory elements and allows these memory elements to intelligently migrate lines amongst themselves depending on the memory requirement of each type of code. A core with a Triangle cache is more suited to execute I/O intensive workloads. We also propose the design of cores that cater to CPU intensive and memory intensive workloads. The area of the TriKon processor is within 2% of a baseline processor, and with such a system, we could achieve a performance gain of 12% for a suite of 16 applications. *This work has been published in the proceedings of the IEEE International Conference on High Performance Computing (HIPC), 2014.*

SOFTWARE DEVELOPMENT

Tejas

Tejas is a cycle accurate architectural simulator that is developed by our research group (Srishti) at IIT Delhi. The simulator has been written entirely in Java, and is shown to be the fastest open-source (Apache v2 license) cycle accurate simulator worldwide (800+ users). I am one of the major contributors (57,000+ lines of code) to this project. The primary contributions include:

- **Translation Engine** : Implemented the translation engine of Tejas. The translation engine converts instructions from x86 to VISA. VISA is the ISA of Tejas; it abstracts the idiosyncrasies of a complex ISA like x86 and is sufficient for timing simulation.
- **Communication Channel** : Designed and implemented the plug and play communication channel of Tejas. It provides seamless support for emulators such as Pin, Qemu, and GPGPU over different communication channels such as shared memory, network, and files.
- **Memory System** : Designed and implemented the cache subsystem (MSHR, coherence, interface with chip interconnect) of Tejas along with a fellow Ph.D. student, Rajshekar Kalayappan.

QemuTrace

Qemu is an open source emulator that can run unmodified operating systems. We developed a patch to Qemu called QemuTrace that provides full system execution traces. These traces are important for studying the execution of system intensive workloads in virtualized as well as native environments. My primary contributions include:

- **Instrumentation** : Instrumented Qemu to generate full system execution traces that include an assembly text of instructions executed, load/store addresses, branch outcomes, system calls, interrupts and privilege level switches in x86.
- **Non-intrusive recording** : Ported a research prototype of Qemu record/replay framework to QemuTrace.

TECHNICAL SKILLS

- **Programming Languages**: C, C++, Java, Python, Bash, Perl, TCL, HTML
- **Parallelization Techniques**: OpenMP, MPI, CUDA, POSIX Threads
- **Versioning Systems**: Mercurial, GIT, Bazaar, SVN
- **Operating Systems**: Linux, OpenBSD, FreeBSD, Microsoft Windows, Mac OS, Sun Solaris

AWARDS AND ACHIEVEMENTS

- Was awarded the **Best Poster Award** for our work on “Ethical Hacking of License Managers” at Security and Privacy Symposium, Delhi, 2015.
- Was awarded the **Outstanding Teaching Assistant Award** twice:
 - Data Structures and Algorithms course taken by Dr. Amitabha Bagchi in Semester I, 2015-16 at the Department of Computer Science and Engineering, IIT Delhi.
 - Software Design Practices course taken by Dr. Vinay Ribeiro in Semester-II, 2015-16 at the Department of Computer Science and Engineering, IIT Delhi.
- **GATE** : Ranked 174/107,086 in the GATE-2010 exam (among top 0.2 percentile). Graduate Aptitude Test in Engineering (GATE) is an all-India examination that primarily tests the comprehensive understanding of various undergraduate subjects in computer science.
- **RoboThrob** : Stood second in a robotics event held at Nirma University, Ahmedabad. The competition required us to develop a robot that is constrained by power, size, and weight; and yet is strong enough to displace the opponent’s robot from the playing arena.

PROFESSIONAL SERVICES AND ACTIVITIES

REVIEWER	Conferences: MICRO, ISCA, IPDPS, HiPC, VLSI Journals: TPDS
TEACHING ASSISTANT	UG courses: Data Structures and Algorithms, Computer Architecture, Software Design Practices, Embedded Systems Lab. PG courses: Operating Systems, Special Topics in Hardware Systems.
MENTOR	Along with Dr. Smruti R. Sarangi, I have mentored the thesis of four masters students: Nitin Gupta, Rohan Bhalla, Coca Sai Prajeeth, and Karishma Agarwal (Best M.Tech Thesis Award).

RECENT TALKS

- Delivered a 3 hour tutorial session titled “Tejas: A Java based Versatile Micro-architectural Simulator” at the 49th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO) in October, 2016.
- Gave a research talk titled “Trikon: A Hypervisor Aware Manycore Processor” at the Universidade Federal do Rio de Janeiro (UFRJ) in September, 2015.
- Gave a research talk titled “Tejas: A Java based Versatile Micro-architectural Simulator” at the Universidade do Estado do Rio de Janeiro (UERJ) in September, 2015.

OTHER INTERESTS

Free style swimming (district level champion), Medium distance cycling, Regular blood donation.

PUBLICATIONS

1. *Comparing Operating Systems from an Architectural Point of View* by Prathmesh Kallurkar, Coca Sai Prajeeth, Divya Gautam, Smruti R. Sarangi. ACM Operating Systems Review (OSR) (to be submitted).
2. *SchedTask: A Hardware-Assisted Task Scheduler* by Prathmesh Kallurkar, Smruti R. Sarangi. IEEE/ACM International Symposium on Microarchitecture (MICRO), 2017. (accepted).
3. *pTask: A Smart Prefetching Scheme for OS Intensive Applications* by Prathmesh Kallurkar, Smruti R. Sarangi. IEEE/ACM International Symposium on Microarchitecture (MICRO), Taipei, Taiwan, 2016.
4. *Tejas: A Java based Versatile Micro-architectural Simulator* by Smruti R. Sarangi, Rajshekar Kalayappan, Prathmesh Kallurkar, Seep Goel, Eldhose Peter. International Workshop on Power And Timing Modeling, Optimization and Simulation (PATMOS), Salvador, Brazil, 2015.
5. *Ethical Hacking of License Managers* by Karishma Agarwal, Prathmesh Kallurkar, Siva Krishna Aleti, Smruti R. Sarangi. Security and Privacy Symposium (SPS), Delhi, India, 2015 (**Best Poster Award**).
6. *Trikon: A Hypervisor Aware Manycore Processor* by Rohan Bhalla, Prathmesh Kallurkar, Nitin Gupta, Smruti R. Sarangi. IEEE International Conference on High Performance Computing (HiPC), Goa, India, 2014.
7. *Architectural Support for Handling Jitter in Shared Memory based Parallel Applications* by Sandeep Chandran, Prathmesh Kallurkar, Parul Gupta, Smruti R. Sarangi, IEEE Transactions on Parallel and Distributed Systems (TPDS), Volume 25, Issue 5, 2014.
8. *UsiFe: An User Space Filesystem with Support for Intra File Encryption* by Rohan Sharma, Prathmesh Kallurkar, Saurabh Kumar, Smruti R. Sarangi, International Conference on Software and Computing Technology (ICST), Singapore, 2011.