Parul Sohal

psohal@bu.edu | linkedin.com/in/psohal | scholar.google.com/psohal | (+1) 650-878-7384

Overview

Systems researcher focused on profile-driven resource management for shared-memory systems. Developed runtime mechanisms that analyze hardware performance counter data, detect workload phases, and translate these measurements into resource allocation policies that reduce contention for shared resources across real-time and cloud platforms. My work emphasizes data-driven design and rigorous evaluation, delivering quantifiable improvements in performance, predictability, and resource efficiency. Interested in extending these systems with machine learning models and AI-assisted tools to analyze large profiling datasets and automatically derive resource management policies for modern large-scale workloads.

Areas of Expertise:	Performance Profiling and Optimization, Resource Isolation and Partitioning, Memory Management, Linux Systems Programming, Hypervisors.
Hardware & Systems:	Intel RDT (CAT, MBA), ARM QoS, Cache Partitioning, DRAM Bandwidth Regulation, Performance Monitoring Units (PMUs).
Programming:	C, Python, Bash.
ML/AI & Tools:	PyTorch, Claude Code, Codex, `Perf`, `SystemTap`.

Education

Sep 2017 – May 2026 | **Ph.D. Computer Science**
Boston University
Advisor: Prof. Orran Krieger

Sep 2013 – May 2017 | **B.A. Computer Science and Mathematics**
Lewis and Clark College

Experience

Sept 20 – Present | **Software Engineer Intern**

Red Hat Inc., USA

- Worked closely with Red Hat engineers and research staff to mature and validate my Ph.D. research within Linux, hypervisor, embedded, and cloud environments.
- Assisted in bringing research mechanisms into production-oriented environments, validating them on enterprise hardware and workload traces representative of real deployments.

Selected Projects

Profile-driven Performance Management Framework (ProPer)

Designed and implemented a dynamic, *Profile-driven Performance Management Framework* for multi-tenant cloud platforms that enforces strong performance isolation across shared

memory hierarchies. Developed automated phase-based allocation policies using hardware performance counters and lightweight probes, and integrated enforcement using Intel CAT and a custom userspace DRAM bandwidth regulator, reducing worst-case slowdown under 20-way contention from $6\times$ to $2.5\times$ without requiring hardware overprovisioning. Led and mentored a graduate student and several undergraduate researchers in system design, kernel integration, and large-scale evaluation.

Envelope-aWare Predictive model (E-WARP**)**

Built and deployed an end-to-end, profile-driven resource management framework for heterogeneous embedded systems to enforce strong timing isolation and predictable performance. Integrated a microsecond-resolution hardware counter profiler into a partitioning hypervisor with ‹1% overhead, and developed *Envelope-aWare Predictive model* that significantly reduces pessimism in the predicted worst-case execution time of an application under regulation. Combined software-based regulation with hardware ARM QoS mechanisms to coordinate DRAM bandwidth control across CPUs and accelerators, including empirical modeling of memory-controller saturation to enable regulation-aware performance control.

Unikernel (UKL)

Collaborated with a Ph.D. researcher to develop the initial Linux kernel target for a unikernel-style execution model, enabling a single optimized application to link directly with the kernel and execute with supervisor privileges in a unified address space. Contributed to kernel build system modifications and runtime integration to support direct application–kernel linkage while preserving compatibility with the broader Linux ecosystem. Delivered a lightweight execution environment suitable for both virtualized and bare-metal deployments.

SELECTED PUBLICATIONS

The PROPER **Framework for Profile-driven Performance Management in the Cloud** in *On Going Work*

Abstract: The efficient consolidation of performance-critical workloads along with best-effort applications in data centers is an open challenge. Current approaches to managing performance bottlenecks do not leverage fine-grained information about the applications' interaction with the underlying hardware. We show that leveraging this information unlocks predictive profile-driven system management. Specifically, in this paper, we propose our Profile-driven Performance Management Framework (PROPER) to bring phase-based profile-driven management at the shared memory hierarchy for performance-critical workloads to cloud systems.
We provide a full implementation of our techniques on a commercial platform (Intel 3rd Generation Xeon Scalable Server) and make three key observations. First, our profiling-based policy can capture the application's changing resource requirements over its execution period and provide dynamic resource management policies. Two, acceptable overheads for both profiling and online management of resources are possible. Third, the framework effectively ensures the timeliness of workloads using our policy-defined resource assignments even when other interfering *best-effort* workloads are contending for the limited resources.

Integrating Unikernel Optimizations in a General Purpose OS in *ACM EuroSys 2023, Rome, Italy* [Paper]

Abstract: This paper presents Unikernel Linux (UKL), a path toward integrating unikernel optimization techniques in Linux, a general purpose operating system. UKL adds a configuration option to Linux allowing for a single, optimized process to link with the kernel directly, and run at supervisor privilege. This UKL process does not require application source code modification, only a re-link with our, slightly modified, Linux kernel and glibc. Unmodified applications show modest performance gains out of the box, and developers can further optimize applications for more significant gains (e.g. 26% throughput improvement for Redis). UKL retains support for co-running multiple user level processes capable of communicating with the UKL process using standard IPC. UKL preserves Linux's battle-tested codebase, community, and ecosystem of tools, applications, and hardware support. UKL runs both on bare-metal and virtual servers and supports multi-core execution. The changes to the Linux kernel are modest (1250 LOC).

A Closer Look at Intel Resource Director Technology (RDT) in *ACM Conference on Real-Time Networks and Systems (RTNS) 2022, Paris, France* [Paper]

Abstract: Unarbitrated contention over shared resources at different levels of the memory hierarchy represents a major source of temporal interference. Hardware manufacturers are increasingly more receptive to issues with temporal interference and are starting to propose concrete solutions to mitigate the problem. Intel Resource Director Technology (RDT) represents one such attempt. Given the wide adoption of Intel platforms, RDT features can be an invaluable asset for the consolidation of real-time systems on complex multi- and many-core machines. Unfortunately, to date, a systematic analysis of the capabilities introduced by the RDT framework has not yet been conducted. Moreover, no clear understanding has been matured about the implementation-specific behavior of RDT primitives across processor generations. And ultimately, the ability of RDT to provide real-time guarantees is yet to be established.
In our work, we aim at conducting a systematic investigation of the RDT mechanisms from a real-time perspective. We experimentally evaluate the functionality and interpretability of RDT-aided allocation and monitoring controls across the two most recent processor generations. Our evaluations show that while some features like Cache Allocation Technology (CAT) yield promising results, the implementation of other primitives such as Memory Bandwidth Allocation (MBA) has much room for improvement. Moreover, in some cases, the presented interfaces range from blurry to incomplete, as is the case for MBA and Memory Bandwidth Monitoring (MBM).

Profile-driven Memory Bandwidth Management for Accelerators and CPUs in QoS-enabled Platforms in *Springer Nature Real-Time Systems Journal 2022* [Paper]

Abstract: The proliferation of multi-core, accelerator-enabled embedded systems has introduced new opportunities to consolidate real-time systems of increasing complexity. But the road to build confidence on the temporal behavior of co-running applications has presented formidable challenges. Most prominently, the main memory subsystem represents a performance bottleneck for both CPUs and accelerators. And industry-viable frameworks for full-system main memory management and performance analysis are past due. In this paper, we propose our *Envelope-aWare Predictive model*, or E-WARP for short. E-WARP is a methodology and technological framework to: (1) analyze the memory demand of applications following a profile-driven approach; (2) make realistic predictions on the temporal behavior of workload deployed on CPUs and accelerators; and (3) perform saturation-aware system consolidation. This work aims at providing the technological foundations as well as the theoretical grassroots for truly workload-aware analysis of real-time systems. This work combines traditional CPU-centric bandwidth regulation techniques with state-of-the-art hardware support for memory traffic shaping via the ARM QoS extensions. We make three key observations. First, our profile-driven methodology achieves, on average, 6% over-prediction on the runtime of bandwidth-regulated applications. Second, we

experimentally validate that the calculated bounds hold system-wide if the main memory subsystem operates below saturation. Third, we show that the E-WARP methodology is practical even when applications exhibit input-dependent memory access patterns. We provide a full implementation of our techniques on a commercial platform (NXP S32V234).

E-WARP: **A System-wide Framework for Memory Bandwidth Profiling and Management** in *IEEE Real-Time Systems Symposium (RTSS) 2020, Texas, USA* [Paper]

Abstract: The proliferation of multi-core, accelerator-enabled embedded systems has introduced new opportunities to consolidate real-time systems of increasing complexity. But the road to build confidence on the temporal behavior of co-running applications has presented formidable challenges. Most prominently, the main memory subsystem represents a performance bottleneck for both CPUs and accelerators. And industry-viable frameworks for full-system main memory management and performance analysis are past due. In this paper, we propose our *Envelope-aWare Predictive model*, or E-WARP for short. E-WARP is a methodology and technological framework to: (1) analyze the memory demand of applications following a profile-driven approach; (2) make realistic predictions on the temporal behavior of workload deployed on CPUs and accelerators; and (3) perform saturation-aware system consolidation. This work aims at providing the technological foundations as well as the theoretical grassroots for truly workload-aware analysis of real-time systems. We provide a full implementation of our techniques on a commercial platform (NXP S32V234) and make two key observations. First, we achieve, on average, a 6% overprediction on the runtime of bandwidth-regulated applications. Second, we experimentally validate that the calculated bounds hold if the main memory subsystem operates below saturation. Won *Best Student Paper Award*.

Unikernels: The Next Stage of Linux's Dominance in *ACM SIGOPS HotOS 2019, Bertinoro, Italy* [Paper]

Abstract: Unikernels have demonstrated enormous advantages over Linux in many important domains, causing some to propose that the days of Linux's dominance may be coming to an end. On the contrary, we believe that unikernels' advantages represent the next natural evolution for Linux, as it can adopt the best ideas from the unikernel approach and, along with its battle-tested codebase and large open source community, continue to dominate. In this paper, we posit that an upstreamable unikernel target is achievable from the Linux kernel, and, through an early Linux unikernel prototype, demonstrate that some simple changes can bring dramatic performance advantages.

PEER REVIEW SERVICE

External Reviewer

- Real-Time and Embedded Technology and Applications Symposium (RTAS), 2023
- Euromicro Conference on Real-Time Systems (ECRTS), 2021–2023
- Design, Automation & Test in Europe (DATE), 2021

INVITED TALKS

- **ACM Conference on Real-Time Networks and System (RTNS)**, 2022, Paris, France
  Presented the paper *"A Closer Look at Intel Resource Director Technology (RDT)"*.
- **DevConf.US**, 2021
  Presented experimental findings and system insights that informed the RDT publication.

- **IEEE Real-Time Systems Symposium (RTSS)**, 2020, Texas, USA
  Presented the paper *"*E-WARP*: A System-wide Framework for Memory Bandwidth Profiling and Management"*.

- **DevConf.US**, 2020
  Presented on the overlapping challenges of interference management in real-time and cloud environments, highlighting shared goals of performance isolation and predictable execution under memory contention.

- **Red Hat Research Day**, 2020
  Discussed hardware and software mechanisms for memory partitioning across embedded and cloud platforms, including practical approaches for coordinated cache and DRAM bandwidth management.

- **DevConf.US**, 2019
  Presented a forward-looking talk on managing main memory bottlenecks in modern cloud processors using commodity software and hardware mechanisms.