YIRONG (EFFY) WANG

PhD Student in Computer Science

Personal website
effywang57@gmail.com
+1 339 545 6591
github.com/Effygal
Boston, MA
LinkedIn

RESEARCH INTERESTS

Analytical Modeling: Cache analysis, system simulation and emulation, system profiling and benchmarking, workload characterization, cloud computing.

Computer Systems: Distributed systems, operating systems, tiered storage, systems for AI, AI for systems.

SKILLS

Programming: C, C++, GDB, SPDK, RDMA, Bash, Python, Go, JavaScript, Rust.

Distributed Systems & Cloud: Concurrency, multithreading, REST, AWS, GCP, Azure, Docker, Kubernetes, Ceph, vSAN.

ML & Generative AI: TensorFlow, PyTorch, CUDA, NumPy, Pandas, SciPy, OpenCL, Scikit-learn, GANs, Hugging Face, LLMs, Transformers.

System Simulation: QEMU, gem5, SimPy, Wolfram Mathematica, Matlab.

Formal Methods: NuSMV, SPIN, TLA+, Z3.

SUMMARY

CS PhD candidate specializes in cache modeling, workload characterization, and system benchmarking. Skilled in analytical modeling, system simulation, and generative AI, with ten years of Linux and four years of distributed storage experience.

EDUCATION

08/2021 – Present **PhD Student in Computer Science** **Northeastern University, USA**
Systems Research Group

12/2012 – 08/2014 **Master of Science** **University of Southampton, UK**
Graduated with distinction

09/2008 – 06/2012 **Bachelor of Engineering** **Tianjin Polytechnic University, China**
GPA: 82/100

EXPERIENCE

08/2021 – Present **Computer Systems Research Assistant & Teaching Assistant** **Northeastern University, USA**

- Design and implement realistic stress microbenchmarks.
- Build SPDK/RDMA distributed storage and analyze file-system I/O traces.
- Characterize workloads using hybrid stochastic processes for cache analysis.
- Research in efficient page-table walk algorithms for TLB caches.
- Co-develop VMware vSAN QoS models for SLA-based performance isolation.
- Collaborate on multiple MOC projects with BU Systems.
- Teach system programming, GDB debugging, and OS fundamentals for three semesters.

Analytical Modeling / Caching / Simulations / Distributed Systems / Performance Benchmarking / System Programming

11/2014 – 08/2021 **IT Administrator** **Yunnan University, China**

- Manage user accounts, deploy software, and maintain data backup and recovery processes.
- Administer network infrastructure, servers, and security systems.
- Develop IT policies, monitor system performance, and resolve technical issues.

Sysadmin / Linux / DevOps / CI/CD / Site Reliability

CURRENT PROJECTS

Analytical Modeling

How FIFO and CLOCK Performance Respond to Popularity Skew **Under submission**

The performance of non-LRU or LFU caches is affected by both frequency and recency, but while prior work on replacement policies has treated these factors intuitively, analytic models have not delved deeply into their separate roles. We present a framework to isolate the effects of each dimension, allowing explicit derivation of miss ratio formulas as frequency and recency are varied, and validate our models in simulation, with mean absolute error for FIFO and CLOCK of < 0.3% for synthetic workloads and < 2.3% for production block storage traces.

Our main theorem shows that the CLOCK miss ratio function is Schur-concave in the frequency vector; for fixed recency distribution, miss ratio decreases with increasing frequency skew. Real workloads are not purely IRM, yet contain significant frequency skew; we extend van den Berg and Toswley's proof of the Schur-convexity of FIFO miss ratio to explain why FIFO beats RANDOM on real workloads, while using our proofs to explain why CLOCK often outperforms LRU.

Benchmarking Toolkit

Configurable and Cache-Accurate Trace Generation for Storage Benchmarking **Github link**

2DIO is a trace generator creating cache-accurate stressful I/O workloads. While existing tools are limited to generating traces with well-behaved, concave hit ratio curves, 2DIO produces ones with tunable complex cache behaviors, particularly performance cliffs and plateaus.

Our framework encodes a workload as a compact parameter triplet, capturing both short-term recency and long-term frequency. This parsimonious parameterization allows researchers to easily translate individual adjustments into predictable cache effects across various eviction policies. It enables the parameter space to be "swept" for exhaustive exploration of desired cache behavior, or to mimic real traces by calibrating parameters to match observed behaviors. Once tuned, the triplet allows traces to be replayed at different scales, adapting to various systems under evaluation while preserving cache behaviors. The benefits of 2DIO extend beyond cache accuracy, enabling portable, cross-platform benchmarking, protecting proprietary information, and facilitating secure knowledge sharing across organizations.

QoS Optimization

UniLP: QoS, Fairness and Isolation in One Formula **Github link**

Performance isolation remains challenging in multi-tenant cloud storage. In private clouds like VMware vSAN, virtual disks striped over shared devices cause resource contention, especially during bursts.

We propose a lightweight, non-intrusive resource allocation mechanism that eliminates reliance on single bottleneck controllers or specialized hardware. A unified linear programming formula runs in a feedback loop to solve for the optimal IOPS rate allocation per virtual disk, maximizing throughput under QoS and device capacity constraints. This allocation is then enforced by hypervisor throttling, preserving the FCFS assumption at the storage end.

Simulations show effective I/O throttling, burst traffic isolation at the source, reduced destination congestion, and significantly lower end-to-end latency.

PAST PROJECTS

Distributed System

Emulating a Distributed Object Storage System on ZNSSD **Github link**

Built with modern C++, FilelogKV emulates a distributed object store leveraging ZNSSD's Zone Append semantics. It uses the write pointer as a weak version ID for consistency. FilelogKV builds atop conventional file systems, emulating ZoneLog via file system calls and using logical block addresses as write pointers to explore system behaviors.

Computational Storage

Emulating an ISC-enabled LSM-tree for Read Optimization **Github link**

The LSM-tree lacks optimization for efficient reads and space use. This proof-of-concept ISC-enabled LSM-tree offloads parallel search computations from host to storage devices to enhance read performance and space efficiency.

Computer Vision

Interactive Visualization of 3D Geometry with Errors **Github link**

This is an interactive visualization tool assisting 3D vision researchers in assessing 3D reconstruction outcomes. It displays 3D meshes from different methods and overlays error and semantic heat maps via color encoding.