Zhongrui Chen

201 S Columbia St – Chapel Hill, NC 27514
jcpwfloi@cs.unc.edu • jcpwfloi.com

Education

Ph.D. Computer Science — **Advisor: Benjamin Berg**
University of North Carolina at Chapel Hill — *2022–present*

B.S. Computer Science, B.S. Mathematics
University of North Carolina at Chapel Hill — *2018–2022*

Research Interests

Stochastic Modeling, Queueing Theory, Scheduling, Caching

Publications

Improving Multiresource Job Scheduling with Markovian Service Rate Policies
Accepted to ACM SIGMETRICS 2025 — *12.2024*
Zhongrui Chen, Isaac Grosof, Benjamin Berg

Simple Policies for Multiresource Job Scheduling
MAMA. ACM SIGMETRICS Performance Evaluations Review 2024 — *04.2024*
Zhongrui Chen, Isaac Grosof, Benjamin Berg

Talks

INFORMS APS — **Atlanta, GA, United States**
A New Class of Policies for Multiresource Job Scheduling — *06.2025*

Northwestern University — **Evanston, IL, United States**
A New Class of Policies for Multiresource Job Scheduling — *04.2025*

INFORMS — **Seattle, WA, United States**
Simple Policies for Multiresource Job Scheduling — *10.2024*

SIGMETRICS MAMA Workshop — **Venice, Italy**
Simple Policies for Multiresource Job Scheduling — *06.2024*

Awards

SIGMETRICS 2024 Student Travel Grant
$1750 — *05.2024*

Academic Service

ACM SIGMETRICS 2025 (Fall)
Reviewer

ACM SIGMETRICS 2025 (Summer, Winter), WWW 2024
Ad-Hoc Reviewer

Research Experience

Graduate Research Assistant
advised by Benjamin Berg — *08.2022–Present*

- Develop a class of low-complexity, low-latency, and throughput-optimal policies — Markovian Service Rate (MSR) policies for multiresource job scheduling.
- Extend the MSR to support jobs that are preemptible, non-preemptible, or preemptible with setup costs.
- Analyze the mean response time of MSR policies.
- Validate the performance of MSR policies using traces from Google Borg.
- Create discrete event simulations to evaluate throughput and delay under various scheduling policies.
- Study resource allocation in open-source ClickHouse database.

Undergraduate Honors Thesis
advised by Praneeth Chakravarthula *08.2021–05.2022*

- Implemented numerical methods to reason dynamical systems from limited observations.
- Experimented with physics-aware video interpolation and extrapolation.
- Coded physical simulators to generate dataset for learning.

Undergraduate Research Assistant
advised by Henry Fuchs under UNC Graphics & VR Group *01.2021–06.2021*

- Worked on reconstructing novel views from a single facial image input.
- Implemented differentiable renderer and mesh fitting with PyTorch3d.
- Contributed an variational autoencoder and texture decoder to reconstruct texture maps for novel views.
- Identified the bottleneck, optimized the training pipeline, and fixed memory leaks. Prebuilt and prefetched the dataset and made training 10x faster.

Mentored Research
advised by Jasleen Kaur *08.2020–12.2020*

Congestion Control: Past, Present and Future (Final Report)

- Generalized the limitations and contributions of the congestion control algorithms.
- Dived into the theory that converts congestion control into a socially concave game.
- Researched fairness and scavenger protocol in congestion control.

Teaching Experience

Undergraduate Teaching Assistant: COMP 572 Computational Photography, COMP 524 Programming Languages, COMP 521 Databases and COMP 411 Computer Organization.

Class Projects

COMP 524 Programming Languages
Lisp Interpreter *08.2020–12.2020*

- Coded a lisp interpreter in Java.
- Implemented basic S-expression primitives, lists, atoms, logical operators and conditionals.
- Implemented lambda function expressions, recursions, curry, and stringify.

COMP 790 OS Implementation
64-bit JOS Kernel *01.2020–05.2020*

- Adapted from MIT 6.828, but in 64-bit version.
- Implemented page translation, context switch, trap handler.
- Implemented cooperative multitasking, Copy on Write (CoW), and Inter-Process Communication (IPC).
- Supported pipe, redirection and basic shell functions.
- Coded a network driver based on the Intel 82540EM chip.

COMP 533 Distributed Systems
Distributed Systems *01.2020–05.2020*

- Used Java RMI, asynchronous RPC library, and NIO to enable process communication across multiple computers.
- Implemented Paxos consensus algorithm to ensure consistency.
- Connected to GUI to showcase state synchronization.

COMP 475 Computer Graphics
Basic Graphics Engine *08.2019–12.2019*

- Implemented a C++ library that provides efficient APIs for geometric primitives, scan conversion, clipping, transformations, compositing, texture sampling, gradients, antialiasing, filtering, parametric curves, and geometric stroking.
- Used Intel MMX features to accelerate vector operations.

Programming Languages

Imperative: C/C++, Rust, Java

Functional: Haskell, Lisp

Scripting: Python, JavaScript, Shell (and variants)

Others: LaTeX, HTML5/CSS3, Matlab, Mathematica