

# Yuanzhuo Yang

608-598-7233 | [yyang682@wisc.edu](mailto:yyang682@wisc.edu) | [linkedin.com/in/yuanzhuoyang](https://www.linkedin.com/in/yuanzhuoyang) | [github.com/ShockYoungCHN](https://github.com/ShockYoungCHN)

## EDUCATION

---

### University of Wisconsin-Madison

Madison, WI

*B.S. in Computer Science, Certificate in Math*

*Aug. 2022 – May. 2025(expected)*

- Overall GPA: 3.97/4.0, Major GPA: 4.0/4.0
- 2022-2024 Dean's List

### Harbin Institute of Technology, Weihai

Weihai, China

*BEng. in Service Science and Engineering*

*Aug. 2020 – Jun. 2022*

- Overall GPA: 88.13/100

## RESEARCH INTERESTS

---

Distributed Systems; Serverless Computing; Operating Systems

## PUBLICATIONS

---

- Yuanzhuo Yang, Kwangjong Choi, Keting Chen, and Tyler Caraza-Harter, "Forklift: Fitting Zygote Trees for Faster Package Initialization", 10th International Workshop on Serverless Computing (WoSC 10), Hong Kong, 2024 (to appear)

## RESEARCH EXPERIENCE

---

### Zygote Tree algorithm | *Go, Python, Docker, AWS*

Apr. 2023 – Jan. 2024

- advisor: Instructor Caraza-Harter, Tyler R.
- Developed an algorithm to generate Zygote Tree, which preloads Python modules based on package dependencies and usage patterns, significantly reducing serverless startup latency.
- Proposed multi-package per node optimization towards the algorithm that significantly boost throughput and minimize tail latency.
- Integrate Zygote Tree with the serverless platform OpenLambda. Improve its throughput to 4.3× and reduce p-95 latency by 2.5× at the cost of 1.2× of memory consumption.
- Performed performance analysis on container primitives, e.g. cgroup and unshare. identified the bottlenecks and bypass them for accurate evaluations.

### RDMA Database | *RDMA, C++*

July. 2024 – Now

- advisor: Prof. Li, Huaicheng; Monga, Sumit
- Collaborated on the development of a general-purpose, object-level RDMA transaction system that bridges the gap between specialized and general-purpose solutions.
- Identify two bottlenecks: message congestion between cache/coordinators and objects reclamation.
- Implemented database benchmarks (TATP, Smallbank, TPCC), constructing Object ID from primary key to read table rows in one RTT, doubling TPCC throughput from 22K to 45K transactions per second on 8-thread compared with the art.

## TEACHING EXPERIENCE

---

### Big Data Systems Peer Mentor | *Python, Docker, GCP*

Sept. 2023 – Dec. 2023

- Assist over 50 students with debugging issues in big data systems and Linux environments.
- Enhanced my understanding of big data systems, including HDFS, Spark, Kafka and Cassandra, through practical experience with GCP.

## PROJECTS

---

**Distributed Systems (MIT 6.824) | *Go***

June. 2023 – Sept. 2023

- Finished MapReduce, Raft, Fault-tolerant K/V service projects, excelling in rigorous testing scenarios, with the Raft implementation notably passing ~2000 tests.
- Delved into core distributed systems papers, including Raft, Time-Clocks in a Distributed System, MapReduce, and FaRM.

## PROGRAMMING SKILLS

---

**Languages:** Go, Python, C++, Rust, C, Java, SQL, JavaScript, HTML/CSS.

**Tools:** Vim, Git, Docker.

**Middlewares:** MongoDB, Kafka, Spark, Redis.