Hankun Dai

Beijing, China | daihankun19@mails.ucas.ac.cn | Personal page | Github profile | Gitee

Education

University of Chinese Academy of Sciences (UCAS) Aug 2023 - Now

- *Master in Computer Science*, GPA: 3.96/4.0

University of California, Berkeley Aug 2022 - Dec 2022

- *Visiting Student*, GPA: 4.0/4.0

University of Chinese Academy of Sciences (UCAS) Aug 2019 - July 2023

- *Bachelor in Computer Science*, GPA: 3.75/4.0

Research Interests

Computer Systems: architecture, parallel/distributed computing, operating system, accelerators, hardware/software co-design

Machine Learning: reinforcement learning, infrastructure

Always interested in systems to boost ML and ML algorithms to design better systems

Projects

Systems

Alpha-Distributed In progress

Searching for better distributed algorithms automatically by AlphaZero-like agents. Human designed algorithms often require unique observations and are not memory/cache friendly. I'm leading it and trying to overcome these challenges. Please stay tuned.

- Model algorithm design as game play
- Meet resource constraints (memory, power)
- Lower latency/cost

Pipelined CPU able to boot Linux on FPGA [Git repo]

Built a RISC CPU on FPGA board and ran Linux core on it. Personal contributions:

- Basic structure of the pipeline and cache
- Support for privilege instructions (CSR registers, TLB, sketchpad extension)
- Interaction with AXI bus
- Prior to that, I also built a simple CPU with an agile, automatic CI/CD framework

Operating system running on dual core RISC-V RocketChip [Git repo]

Designed a Unix-like OS running on a dual-core RISC-V CPU on FPGA. I wrote all features/components, including:

- Interactive bootloader
- Exception handler, process scheduler and task binding
- Process communication and semaphore
- Virtual memory management
- File system
- Device drivers for communication on NIC

Tiny C compiler based on ANTLR [Git repo]

Implemented a compiler with multiple translation stages for core features of C. Syntax analysis was done by ANTLR and the target architecture was RISC-V 64GC. I did most of the work, including:

- Wrote some of the grammar rules
- Semantic analysis

- Intermediate representation generation
- Vector operation extension to C
- Assembly generation and optimization

In-memory database query engine [Git repo]

We developed a query engine (including a planner) for a memory database system used in research. Besides relational operations and SQL, it supported advanced features like prefetching. Personal contributions:

- OLAP support (columnar store, scan, index scan)
- Filter and project operator implementation
- SQL aggregate keyword (group, order by)

Machine Learning

Reimplementation and evaluation of ECCV NeRF [Git repo]

- Reimplemented the work *NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis* with PyTorch

Survey and improvement of TVCG PRS-Net, *advised by Prof. Lin Gao* [Git repo]

- Reimplemented the work *PRS-Net: Planar Reflective Symmetry Detection Net for 3D Models* and replaced linear transformations with high-performance ones
- Proposed several optimizations for higher accuracy and generalization on other datasets

Thesis

Optimization and Acceleration of Atmospheric Model Core FV3 on AMD GPUs, *advised by Prof. Huiyuan Li*

FV3 is the core of Global Forecast System (GFS) simulation suite. This thesis acts as part of a system software development project. We did a survey on its principles and analyzed effects of some numerical methods. We ported critical parts of it to AMD GPUs and optimized their performance on distributed systems.

Awards and Competitions

Outstanding Student of UCAS (top 20%) 2020 - 2021

Academic Scholarship of UCAS 2020 - 2021

The first Tsinghua Jittor's AI challenge competition Spring, 2021

- Award for entering B ranking list (50 teams)

Related Coursework

Undergraduate:

Computer Organization (where we did a survey on side channel attacks, selected as presentations),
Computer Architecture, Operating System, Compiler: Principles and Practices, Intro to AI

Graduate:

Computer Networks (where we built a tiny HTTP server),
GPU Programming and Its Architecture (where we wrote and optimized a yolo-v3 net with CUDA),
AI Computing Systems, Advanced Computer Architecture, Parallel Computing

Skills

Programming languages: C/C++, Python, CUDA, Verilog, Assembly, Mathematica, Java

TOEFL: 101 (> 2 years ago, will be renewed)

Software and Tools: MPI, PyTorch, Visual Studio (Code), Xterm, Git, Vivado, Qemu