Ruobing Han

Email : hanruobing@gatech.edu Homepage : https://drcut.github.io/

Education

Georgia Institute of Technology(GT) 2021.5 – Now, USA
PhD Candidate , Computer Science (CS)
Research Area: compiler, architecture, ML system
Advisor: Prof. Hyesoon Kim
Peking University(PKU) 2014.9 – 2018.7, China
Bachelor of Science , Computer Science (CS)

Internship

Google Sunnyvale, CA, USA

- **ML Debug toolkit development (2023 summer intern)**
  - Work in Machine Learning Functional Debugging team
  - Implement plugins for Tensorflow TPU compilation
  - Implement compiler static analysis to detect bugs in Tensorflow programs
- **Compiler Development (2022 summer intern)**
  - Work in LLVM Core team
  - Implement non-trivial LoopUnswitching with LLVM Function Pass
  - Integrate instruction hoist into LoopUnswitching
  - The patches are submitted to Phabricator for reviewing (D127765, D127770, D128001)

Research

Improving Incremental Building Execution Time

- Enhance performance of the incremental build process by recording previous compilation results.
- Develop and implement a proof-of-concept model in the LLVM-14 compiler, achieving a 6.72% speedup on popular C++ projects.
- In the Proceedings of The International Symposium on Code Generation and Optimization (CGO) 2024.

Solving the Phase-Ordering Problem with Reinforcement Learning

- Develop a Reinforcement Learning model to address the phase-ordering problem.
- Propose a novel pruning solution that exponentially expands the search space, enabling the Reinforcement Learning model to find an optimal solution in a reasonable time frame.
- Our solution generates programs that are 12% faster or 17.6% smaller than the programs produced by LLVM O3/Oz optimizations.
- In the Proceedings of The International Conference on Compiler Construction (CC) 2024.

Porting CUDA to the x86 architecture

- Build a framework to execute CUDA source code with the latest features on CPU devices.
- Implement a transformer to translate CUDA's SPMD kernels into MPMD+SIMD format based on LLVM.
- Improve the coverage from 68% (previous projects) into 90% on CUDA SDK 10.0 samples.
- In the Proceedings of ACM Transactions on Architecture and Code Optimization (TACO).

- Github repo: `https://github.com/cupbop/CuPBoP`

Low-precision distributed training Neural Network

- Propose an algorithm to avoid overflow while using low-precision floating-point for gradients.
- Use 8-bit floating-points to train ResNet50 on large scale distributed system.
- In the Proceedings of the International Conference on High Performance Computing 2021.

PUBLICATION

Conferences

- **Ruobing Han**, Hyesoon Kim. "Exponentially Expanding the Phase-Ordering Search Space via Dormant Information" The International Conference on Compiler Construction (CC) 2024.
- **Ruobing Han**, Jisheng Zhao, Hyesoon Kim. "Enabling Fine-Grained Incremental Builds by Making Compiler Stateful" The International Symposium on Code Generation and Optimization (CGO) 2024.
- **Ruobing Han**, James Demmel and Yang You. "Auto-Precision Scaling for Distributed Deep Learning" International Conference on High Performance Computing 2021.

Journals

- **Ruobing Han**, Jaewon Lee, Jaewoong Sim, Hyesoon Kim. "COX: Exposing CUDA Warp-Level Functions to CPUs" ACM Transactions on Architecture and Code Optimization (TACO) 2022.
- Peng Sun, Wansen Feng, **Ruobing Han**, Shengen Yan and Yonggang Wen. "Optimizing Network Performance for Distributed Deep Neural Network Training on GPU Clusters: ImageNet/AlexNet Training in 1.5 Minutes." IEEE Transactions on Big Data 2020.

Open Source Project Contribution

- **CuPBoP**
  - Support executing NVIDIA CUDA programs on non-NVIDIA devices (e.g., CPUs, AMD GPUs);
  - Corresponding papers are accepted by TACO2021.
  - Project: `https://github.com/cupbop/CuPBoP`
  - Star: 40
- **Vortex GPU**
  - Support hardware and software prefetch.
  - Writing the tutorial for developers.
  - Corresponding project, workshop and tutorial was held on MICRO2021.
  - Project: `https://github.com/vortexgpgpu/vortex`
  - Star: 921.
- **OpenMMlab**
  - Support converting Detection/Segmentation/Editing models from Pytorch into ONNX.
  - Project: `https://github.com/open-mmlab`
  - Star: 20K+