

KONRAD HANDRICK

✉ konradha@yahoo.com

🐙 github.com/konradha

🌐 [linkedin.com/in/konradha](https://www.linkedin.com/in/konradha)

🌐 [konradha.com](https://www.konradha.com)

Based in Switzerland

Technical Strengths

Languages: Python, C++, C, Fortran

Systems: Linux, CUDA, MPI, OpenMP, LLVM, SLURM, Flux

Performance & Architecture: profiling/benchmarking, memory hierarchy, vectorization, distributed computing

Numerics & Algorithms: numerical linear algebra, PDEs & nonlinear waves, optimization, advanced data structures & algorithms

ML & Infra: PyTorch, vLLM, compilers, accelerators

Experience

Founder

Jan 2026 – Jul 2026

Stealth

Switzerland/Poland/UK

- Data stack for persistent maritime situational awareness. Computational ocean acoustics, audio ML
- *Stack:* C++, Pytorch, NixOS, acoustics ML, MCP

ML Infrastructure Engineer, part-time

Oct 2025 – current

stepping stone AG

Bern, Switzerland

- Profiled GPU utilization and inter-node communication patterns to identify bottlenecks in memory bandwidth, kernel launch overhead, and scheduling; knowledge transfer about Language Model Arithmetic to teams of engineers.
- *Stack:* vLLM, NCCL, PyTorch, Linux, Golang

Computing Graduate Scholar

May 2025 – Sept 2025

Lawrence Livermore National Laboratory

Livermore, CA

- Enabled compilation for batched point-to-point communication ops in `torch.compile`, fixing distributed variable handling in Dynamo tracing to support high-throughput multi-GPU inference and training (PyTorch PR #161213, part of May/June 2026 release).
- Worked close to the metal in a large C++/Python codebase, reasoning about memory layout, concurrency, and communication/computation overlap on GPU clusters.
- *Stack:* PyTorch, C++, CUDA, NCCL, MPI, distributed systems

ML Compiler Engineer, Intern

June 2024 – Nov 2024

Axeler AI

Zurich, Switzerland

- Implemented proprietary datatypes and compiler ops for an in-memory computing architecture, paying close attention to data layout, quantization, and numerical stability.
- Extended the TVM compiler stack for a RISC-V based ML accelerator, working on code generation, scheduling, and low-level performance tuning.
- *Stack:* TVM, C++, Python, RISC-V, PyTorch

Compiler Engineer, Contract

June 2022 – Nov 2022

QuantStack

Berlin, Germany

- Core contributor to LFortran, a modern open-source Fortran compiler targeting LLVM.
- Built a robust frontend parser supporting 500k+ LOC of SciPy's F77 codebase; identified and fixed abstract syntax tree and ASR generation issues in real-world numerical code.
- Work featured in the LFortran blog post "LFortran compiles 60% of SciPy".
- *Stack:* LLVM, Fortran, C++, Python

Compiler Engineer, Intern

Oct 2021 – April 2022

AMD (formerly Silexica)

Cologne, Germany

- Developed LLVM-based compiler middleware for an FPGA backend, including data-shape optimization and allocation passes sensitive to memory bandwidth and on-chip resource constraints.
- Worked in a Linux/C++ environment with a strong focus on predictable performance and understanding how hardware decisions impact end-to-end latency.
- *Stack:* LLVM, C++, HLS, Python, Linux

Education

ETH Zurich

2025

M.Sc. Computational Science & Engineering

Zurich, Switzerland

- Thesis: *Data generation for Foundation Models for Scientific ML*.
- Graduate coursework spanning numerical analysis, high-performance computing, physics and discrete optimization.

ETH Zurich

2021

B.Sc. Computational Science & Engineering

Zurich, Switzerland

- Bachelor thesis in designing a fast offloading engine for serverless workflows (RISC-V, qemu).