GPU Track Reconstruction Optimization Timeline

November 2024 – January 2026

2024.11–2025.1

- Established the traccc (GPU track reconstruction) environment

2025.2–2025.3

- Studied traccc code and algorithms (over 70,000 lines)
- Analyzed next steps and reported to the European team
- Revised the manuscript for YunChen's paper

2025.4

- Set up the Nvidia Nsight Systems (profiler) environment
- Profiled traccc to identify bottlenecks
- Created figures for YunChen's paper

2025.5

- Analyzed bottlenecks; attempted code modifications and debugging
- Split the fit kernel, increasing throughput by **10%**
- Assisted YunChen with the **VLSICAD** (conference) submission

2025.6

- Replaced Kalman gain matrix (track fitting computation) operations with INT8 MLP
- Achieved **186%** speedup but observed physics accuracy degradation
- Reported results to the European team
- Assisted YunChen with the **TJCAS** (conference) submission
- Attempted Nsight Compute (kernel profiler) setup (severe environmental issues)

2025.7

- Prepared slides & scripts (EN/CN) for YunChen's **VLSICAD 2025** oral
- Prepared slides & scripts for the **TJCAS** oral presentation
- Successfully established the Nvidia Nsight Compute (NCU) environment

2025.8

- Created posters for **TJCAS** and **FastML** (workshop)
- Attended **VLSICAD** and **TJCAS**

— Continued on next page —

2025.9

- • Attended **FastML**; started planning for Weak Lensing competition
- • Assisted Edwin with NCU profiling of Token Reduction
- • Established basic NCU Profile Flow Optimization for large profile data
- → *Begin: Ongoing NCU Profile Flow Optimization*

2025.10

- • Put coursework aside for Weak Lensing competition (solo sprint)

2025.11

- • Continued Weak Lensing sprint until mid-November; caught up on coursework
- • Conducted NCU profiling on traccc; identified potential for batching

2025.12

- • Completed multi-event batching optimization (mid-month)
- • Achieved **93%** speedup with no physics accuracy loss
- • Researched next steps; attempted several incorrect approaches

2026.1

- • Implemented conditional Jacobian matrix aggregation (**18%** speedup, no degradation)
- • Observation: Batching shifted find/fit (track finding and fitting) from memory-bound to compute-bound
- • Analysis: FPGA infeasible for high FLOPS tasks (strict FP64 requirements)
- • Future: Thesis focuses on GPU register pressure optimization
- → *End: NCU Profile Flow Optimization*

Legend

• Technical/Optimization • Conference/Paper • Collaboration/Other

Green Badge = Performance Achievement Red Text = Accuracy Concern Blue Sidebar = NCU Flow Optimization Period

---

Key Performance Achievements

10%	Fit kernel splitting	2025.5
186%	INT8 MLP replacement (with accuracy trade-off)	2025.6
93%	Multi-event batching (no accuracy loss)	2025.12
18%	Conditional Jacobian aggregation (no accuracy loss)	2026.1

Generated: January 2026