

# Plans for FIDIUM at Goethe University

14/04/2023

V. Lindenstruth, G. Kozlov, A. Redelbach

# Ongoing work and developments for FIDIUM – AP-1

Efficient utilisation of multi-core CPU and GPU resources in compute intensive workflows

Set of benchmarks for CPU/GPU performances in reconstruction tasks

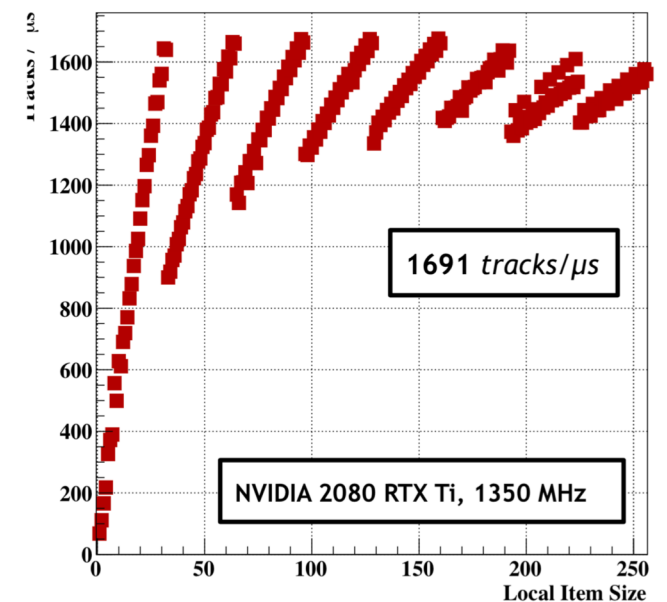
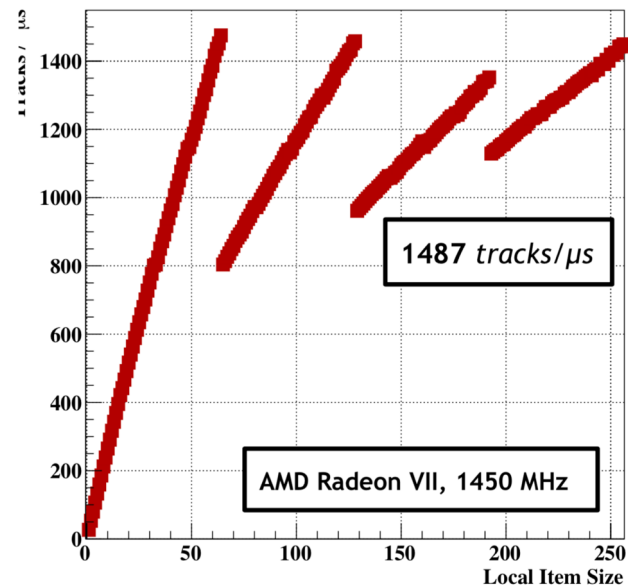
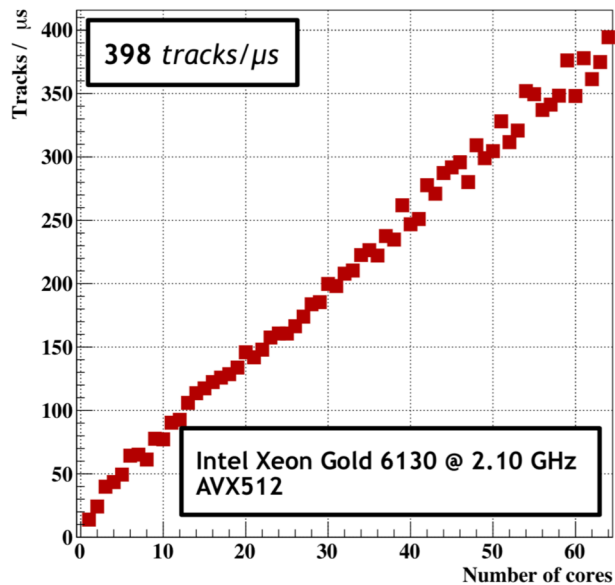
→ Integration into other computing environments

→ GPU implementation / common CPU/GPU solution XPU framework <https://github.com/fweig/xpu>

→ Device code is compiled once for each backend (CUDA, HIP, OpenMP), device selection at runtime

→ Extending the functionality of XPU : integration of SIMD intrinsics

Scaling of track fitting



# Ongoing work and developments for FIDIUM – AP-2

Detailed comparisons between XCache and Disk Caching on the Fly

→ Developments based on XCache (with direct cache access)

Developments and tests for FAIR data lake prototype

→ Study of use cases for efficient deployment of dynamic caching in terms of most relevant parameters

Performance tests for efficient data access for high-bandwidth WAN (between Goethe HLR and GSI): Performance gains for for dynamic caching based on shared storage for increasing number of nodes

→ Goal: Better coverage/understanding of multi-dimensional parameter space for dynamic caching for different use cases


→ Efficient mechanisms for data placement and replication (e. g. hash-based)

# Focus: Aspects of green computing

Significant emissions from the operation of data centres and high-performance computing facilities, requiring increasingly energy-efficient facilities

Reductions in green house gas (GHG) emissions can also be achieved with current technology, through a combination of optimization of resource use and careful planning of timing and siting of computation.

Tools available to **calculate the carbon footprint of computations** run on a HPC platform.



Running a singularity job at a node at the GSI Green Cube would be beneficial from the perspectives of energy saving.

Goal: Estimate the GHG emissions impact of a particular job  
With knowledge of the hardware, energy mix, and particulars of the job request, a reasonably accurate range of possible GHG emissions should be possible

The energy of an executed algorithm depends on factors such as the running time, the number/type of computing cores (CPU or GPU), the amount of memory mobilised and the power draw of these resources

See e.g. Climate impacts of particle physics, Snowmass 2021, <https://doi.org/10.48550/arXiv.2203.12389>

Green Algorithms: Quantifying the carbon footprint of computation, <https://doi.org/10.48550/arXiv.2007.07610>

# Focus: Efficient Machine learning for resource utilisation

Resource management characterizes real-time decisions of scheduling, allocation and migration of user applications to various virtual/physical nodes for the purpose of computation and execution.

**Proactive** approaches of traffic management rely on the prior knowledge of the forthcoming workload/applications and workload balancing

**Reactive** resource management handles the traffic load at actual arrival by allocating resources and corresponding job scheduling

**Workload estimation and analysis** can efficiently be done by **neural networks**

Optimization e.g. via energy consumption, execution time and possibly entropy of data traffic

→ Some neural network-based solutions exist, have to be applied and optimized for FIDIUM use cases

# Main topics

Further developments of C++ library for GPU software development

Efficient utilisation of multi-core CPU and GPU resources in compute intensive workflows

Studies for carbon footprint of FIDIUM computations run on a HPC platform

Workload estimation and analysis by neural networks

Optimised load balance: decision where to execute a certain algorithm on the current and predicted load of available CPUs and GPUs