Scientific computing

Activities of the ZPPT group

Stefan Schaefer

APPP workshop, October 7th 2025

HELMHOLTZ

Scientific computing in theoretical particle physics

Scientific computing can mean many things and has spread considerably over the last decades.

- Use of significant computer resources to achieve scientific goals.
- Develop new methods to use resources efficiently → reduce carbon footprint

For the **ZPPT** group this means two things

Lattice Gauge theory
- Simulate QCD on a lattice → software, algorithm development
- Data management

Perturbation theory
- Evaluation of Feynman integrals using computer algebra.

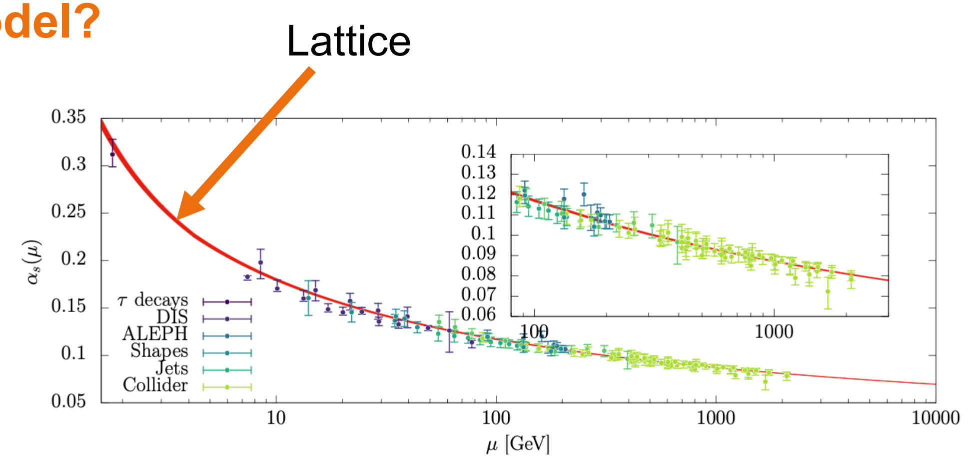
Standard model physics

Why are we using computers to learn about the Standard Model?

The standard model of particle physics is a great success.

To make this statement, one needs to know what it predicts.

Many tests involve strong interaction physics
- collide protons
- testing to high precision as in $(g-2)_\mu$
- several highlights during this workshop.



ALPHA coll, arXiv:2501.06633

To know the SM result one needs
- High precision perturbation theory (involving millions of Feynman diagrams)
- First principles strong interaction quantities → lattice computations

Precision theory predictions are necessary to test the standard model to new levels.
→ Standard model QCD parameters (quark masses, α_s,…) come now from ``pure'' theory (lattice).

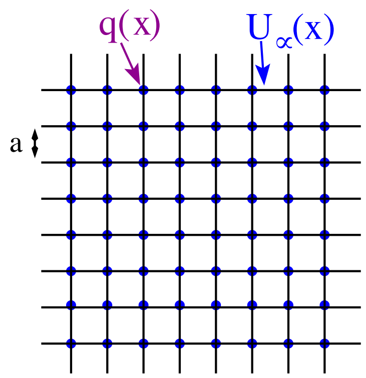
Theory computations are much cheaper than experiments.
One should not use experimental data for what can be computed in theory (e.g. α_s).

Lattice Field Theory

Non-perturbative quantum chromodynamics

Discretized space-time → discrete path integral → Markov Chain Monte Carlo with 10^{10} degrees of freedom.

$$\langle A \rangle = \frac{1}{Z} \int dU\, e^{-S(U)}\, A(U) \cong \frac{1}{N} \sum_{i=1}^{N} A(U_i)$$



These degrees for freedom all interact.

Need:

- Massively parallel computers to perform necessary computations.
  → Distribute the lattice and related computation on many processors.
- Develop algorithms that efficiently use the available hardware.
- Manage sizeable amounts of data according to FAIR data principles.

Key algorithms

- Markov Chain Monte Carlo is based on the Hybrid (Hamiltonian) Monte Carlo algorithm.
- Heavy development in highly efficient solvers for linear equations.
- Solvers for differential equations.

Long tradition linking to applied math. Recent example DFG research unit FOR5269.

CLS

An example of an effort

Generation of gauge field configurations is shared among many researchers, sometimes organized as collaborations, sometimes more loosely as in CLS.

Precision results need fine and large lattices.
Typical values $a = 0.05\mathrm{fm}$ and $L = 6\mathrm{fm} \rightarrow \frac{L}{a} = 128$

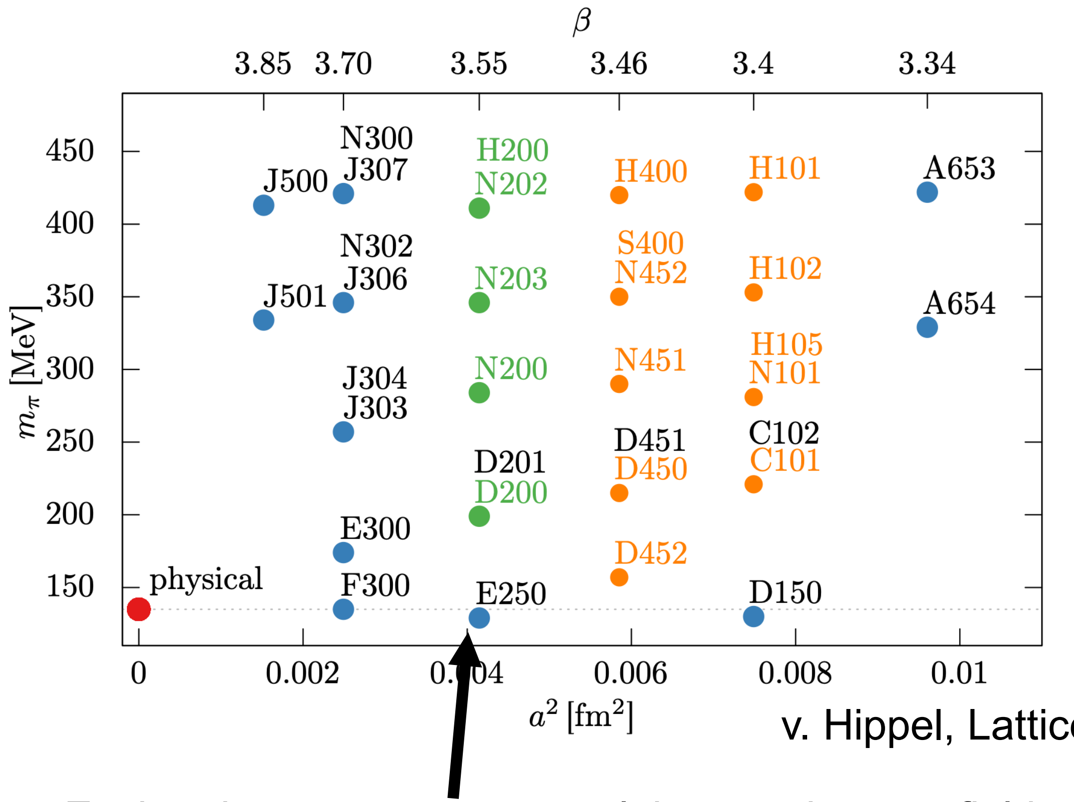
Good planning is essential to sustainability of this kind of effort.
→ Human time
→ CO_2 emmissions
→ Responsible use of tax payer's money

1 configuration (E250) 750 €, 550kg CO_2 ….need 1000s of those.

Gauge field configurations are being used many different projects.

→ Need for good data management.

Ensembles generated by **C**oordinated **L**attice **S**imulations



v. Hippel, Lattice 24

Each point represents several thousand gauge fields.
Each ensemble in the lower row cost $n \times 1$M€.

Simulation of QCD with $N_f = 2 + 1$ flavors of non-perturbatively improved Wilson fermions #3

Mattia Bruno (DESY, Zeuthen and NIC, Zeuthen), Dalibor Djukanovic (Helmholtz Inst., Mainz), Georg P. Engel (INFN, Milan Bicocca and Milan Bicocca U.), Anthony Francis (Helmholtz Inst., Mainz), Gregorio Herdoiza (Madrid, Autonoma U. and Madrid, IFT) et al. (Nov 14, 2014)

Published in: *JHEP* 02 (2015) 043 • e-Print: 1411.3982 [hep-lat]

pdf DOI cite claim reference search 342 citations

Changing paradigm

From CPU based computers to interconnected GPUs

Past to present
Many thousands of CPU nodes, connected by Infiniband.

Example JUQUEEN (2012)
458.752 IBM PowerPC-A2-Cores
5.9 Petaflops
2.3 Gflops/Watt

Present to future
Thousands of GPU nodes, connected by Infiniband.

Example JUPITER (2025)
24.000 Nvidia GH 200 GPUs
1000 PetaFlops
73 Gflops/Watt

x170

x 32

GREEN500 LIST - JUNE 2024

R_{max} and R_{peak} values are in PFlop/s. For more details about other fields, check the TOP500 description.

R_{peak} values are calculated using the advertised clock rate of the CPU. For the efficiency of the systems you should take into account the Turbo CPU clock rate where it applies.

Green500 Data

← 1-100 101-200 201-300 301-400 401-500 →

Rank	TOP500 Rank	System	Cores	Rmax (PFlop/s)	Power (kW)	Energy Efficiency (GFlops/watts)
1	189	**JEDI** - BullSequana XH3000, Grace Hopper Superchip 72C 3GHz, NVIDIA GH200 Superchip, Quad-Rail NVIDIA InfiniBand NDR200, ParTec/EVIDEN EuroHPC/FZJ Germany	19,584	4.50	67	72.733
2	128	**Isambard-AI phase 1** - HPE Cray EX254n, NVIDIA Grace 72C 3.1GHz, NVIDIA GH200 Superchip, Slingshot-11, HPE University of Bristol United Kingdom	34,272	7.42	117	68.835
3	55	**Helios GPU** - HPE Cray EX254n, NVIDIA Grace 72C 3.1GHz, NVIDIA GH200 Superchip, Slingshot-11, HPE Cyfronet Poland	89,760	19.14	317	66.948

https://top500.org

Source: FZ Jülich

Source: FZ Jülich

Porting to GPU hardware

GPU hardware is very different from CPUs

Few powerful cores on a CPU
→ Thousands of cores on a GPU

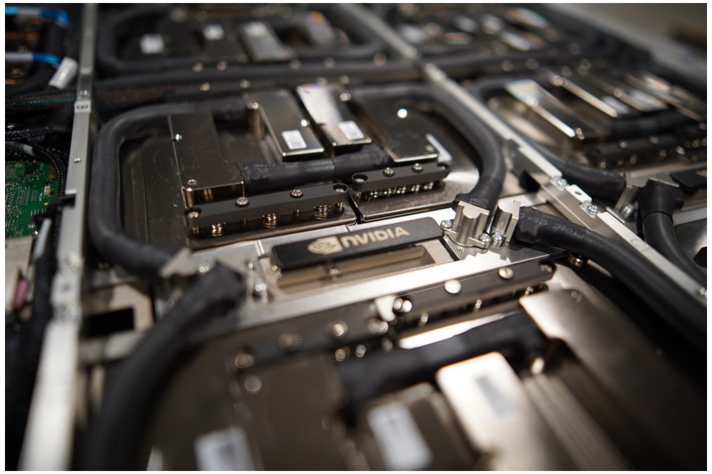
GPUs are rather specialized and different from CPUs
→ Need to rewrite large fraction of the code.

On a CPU we had for each core one thread (or even one MPI rank) running.

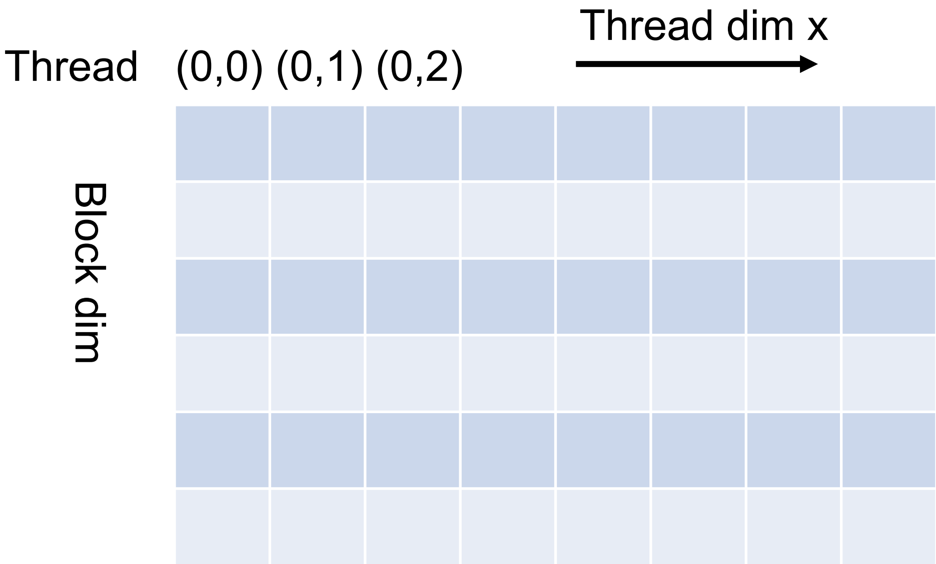
GPU: thousands of threads, acting in blocks
→ Strong hierarchy, need for change in algorithms.

Need to be very careful how to organize the memory access.
→ Specific memory layout of the fields
→ Organize computation for coalesced access.

Memory access is virtually always the limiting factor.



Source: FZ Jülich



openQCD on GPUs

Status

openQCD is a widely used code to simulate QCD on the lattice. (CLS, Fastsum, PACS, …)

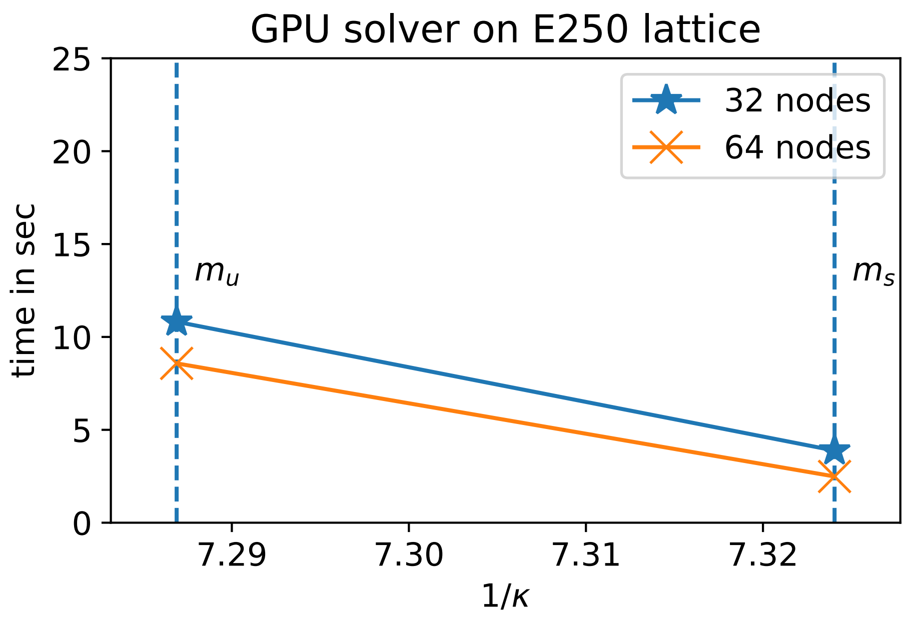
Started in 2010 by M. Lüscher and S.S.

- Generation of gauge fields.
- Efficient multigrid solver for the Dirac equation.
- Reduced communication between nodes.
- Highly efficient on CPU based machines.
- Many state of the art algorithms implemented.

Porting to GPUs by A. Rago (Odense) and S.S.

- Complete rewrite of all computational routines in CUDA.
- Full solver suite for the Dirac equation available since 2024.
- Significant performance gains during the last year.

Collaborating with users to port code to new framework.

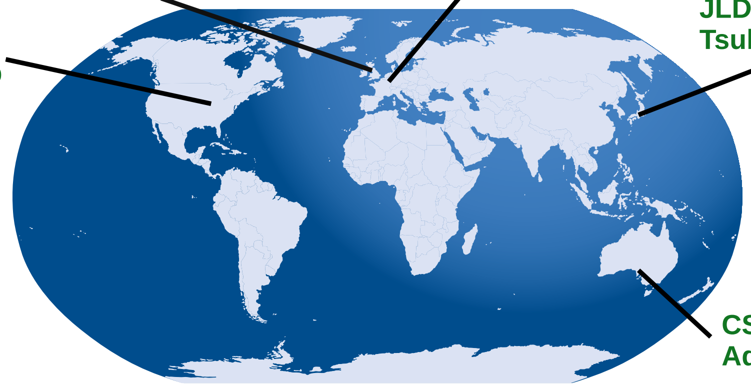


Data

International Lattice Data Grid

Sharing and management of large datasets of gauge field configurations.

- Share data immediately within collaboration.
- Make at later stage available to community.
- A few PB of data for a collaboration.



Need fine-grained access control.
Implementation of FAIR data principles.
Findable, **A**ccessible, **I**nteroperable, **R**eusable

International Lattice Data Grid (since the early 2000s)

Implements a distributed federated infrastructure

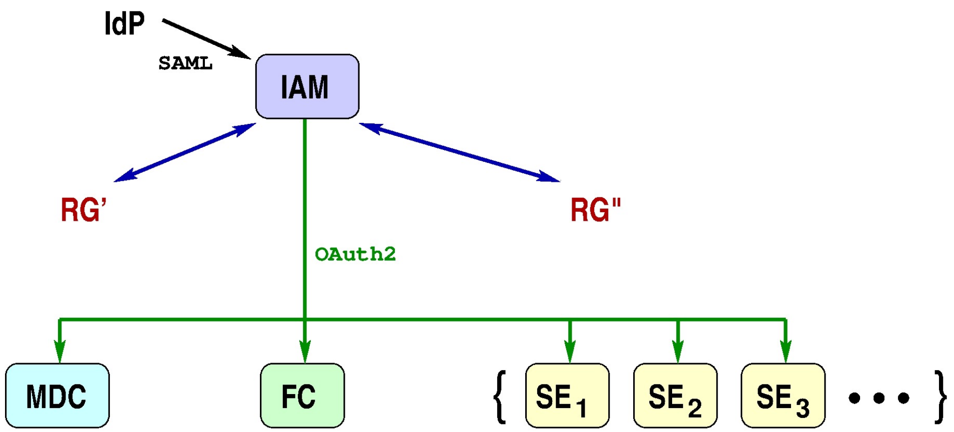
- multiple storage resources (and technologies)
- central uniform user management
- scientific metadata: parameters, provenance
- administrative metadata: storage URL, permissions, formats

Recent efforts within PUNCH4NFDI + DESY have modernized the software.

ILDG

Basic architecture

No monolithic framework
→ ILDG defines a modular and flexible architecture of services



Basic interoperable components

- Storage elements (dCache, StoRM, Teapot, S3, …)
- **I**dentity and **A**ccess **M**anagement
- Metadata Catalogue
  Configurable with multiple schema
  Powerful search language
- File Catalogue
  persistent logical identifiers
  ↕
  variable and multiple physical storage locations

Optional add-on services

- Interfaces for automatic markup or complex search
- Metadata harvesting
- File Transfer Service
- Hierarchical access management

ILDG 2.0

Ready for use in lattice community and beyond

Many communities have similar requirements.

Existing solutions lack one or more important features

- support for large data objects
- rich community-wide agreed metadata schema
- fully token-based and fine-grained access control
- global user management

Potential extension in PUNCH 2.0

ILDG-like data management for framework small to midsize experiments. (ALPS, IAXO)

PUNCH 4NFDI

Synergies with CTAO

CTAO has similar (technical) challenges

- multiple data storage and metadata catalogues with restricted access
- Token-based access control

  Central policy decision vs. distributed services.

  Hierarchical delegation of decisions.
- Central **I**dentity and **A**ccess **M**anagement

  INDIGO IAM: developed by INFN-CNAF

  central for WLCG, SKA, ILDG, CTAO,…

→ Joint IAM Hackathon in July hosted by CTAO and ILDG

Algorithm development

Multilevel algorithms

Aim compute two-point correlation functions.

$$\langle O(x)O(y)\rangle$$

- Standard algorithm uncertainty $\sim 1/\sqrt{N}$

Problem: For large separation signal-to-noise ratio degrades exponentially with $e^{-m\,|x-y|}$.

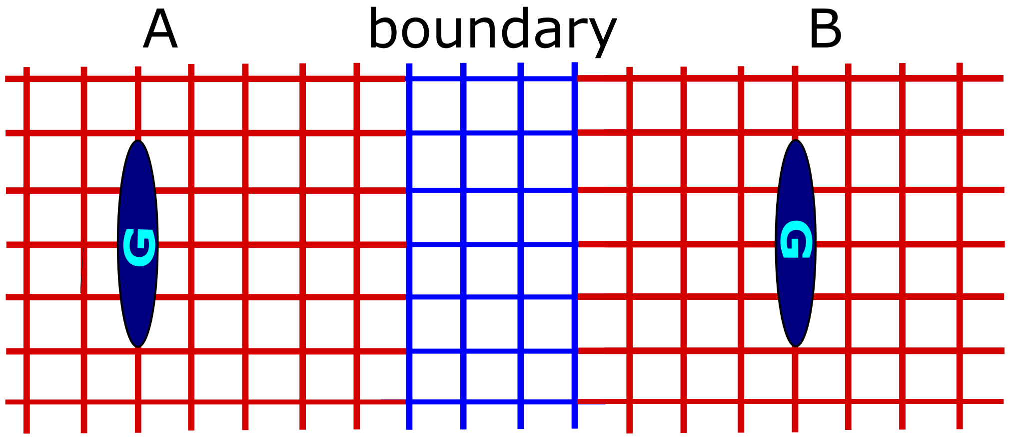
- Use the locality of the theory

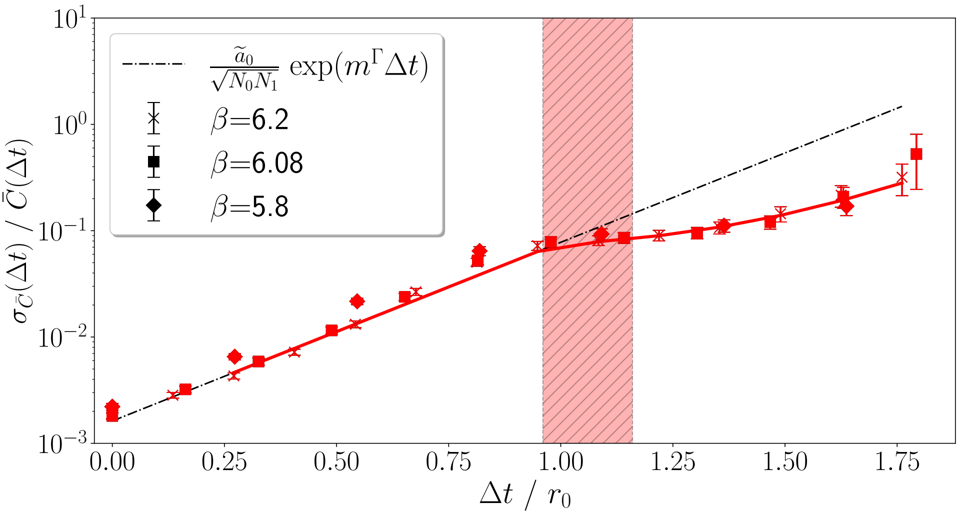
  Improve scaling to (ideally) $\sim\frac{1}{N}$.

We have been able to demonstrate the feasibility of these algorithms.

So far limited ot pure gauge and quenched QCD.

Upcoming: Full QCD simulations.





Barca et al, Phys.Rev.D 110 (2024) 5, 054515

Summary

Possible connections to other groups

Scientific computing is goes both ways

- Use computing to solve challenging science questions.
- Invent methods to make more efficient use of resources → focus on time to solution.

Gained some experience in efficient use of GPUs.

We would be happy to share this. And happy to learn from others.

The lattice community has significant experience with data management.

PUNCH4NFDI effort at DESY has been instrumental to revitalize the ILDG.

The ILDG framework can also work for other communities with similar requirements.

Special thanks to H. Simma for the ILDG material