Kick-off meeting
Spoke 2 Centro HPC

Porting to GPU, experience from Theory-HEP:

porting di codice per calcoli di QCD su GPU

Francesco Sanfilippo, INFN RM3

Istituto Nazionale di Fisica Nucleare

13 Ottobre 2022

LATTICE QCD SIMULATIONS

(see also D.Cosmai at 15:20, Parallel WP1)

First principle simulation of strong interactions

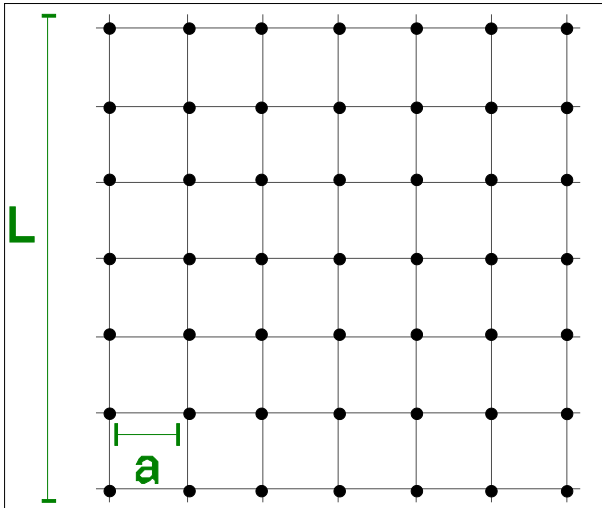
Quantum Chromodynamics on a Lattice

4D (spacetime) with $O(10^{10})$ degrees of freedom

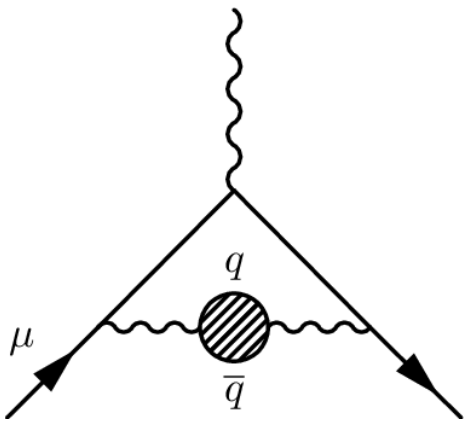
Hybrid Monte Carlo + Molecular Dynamics simulations

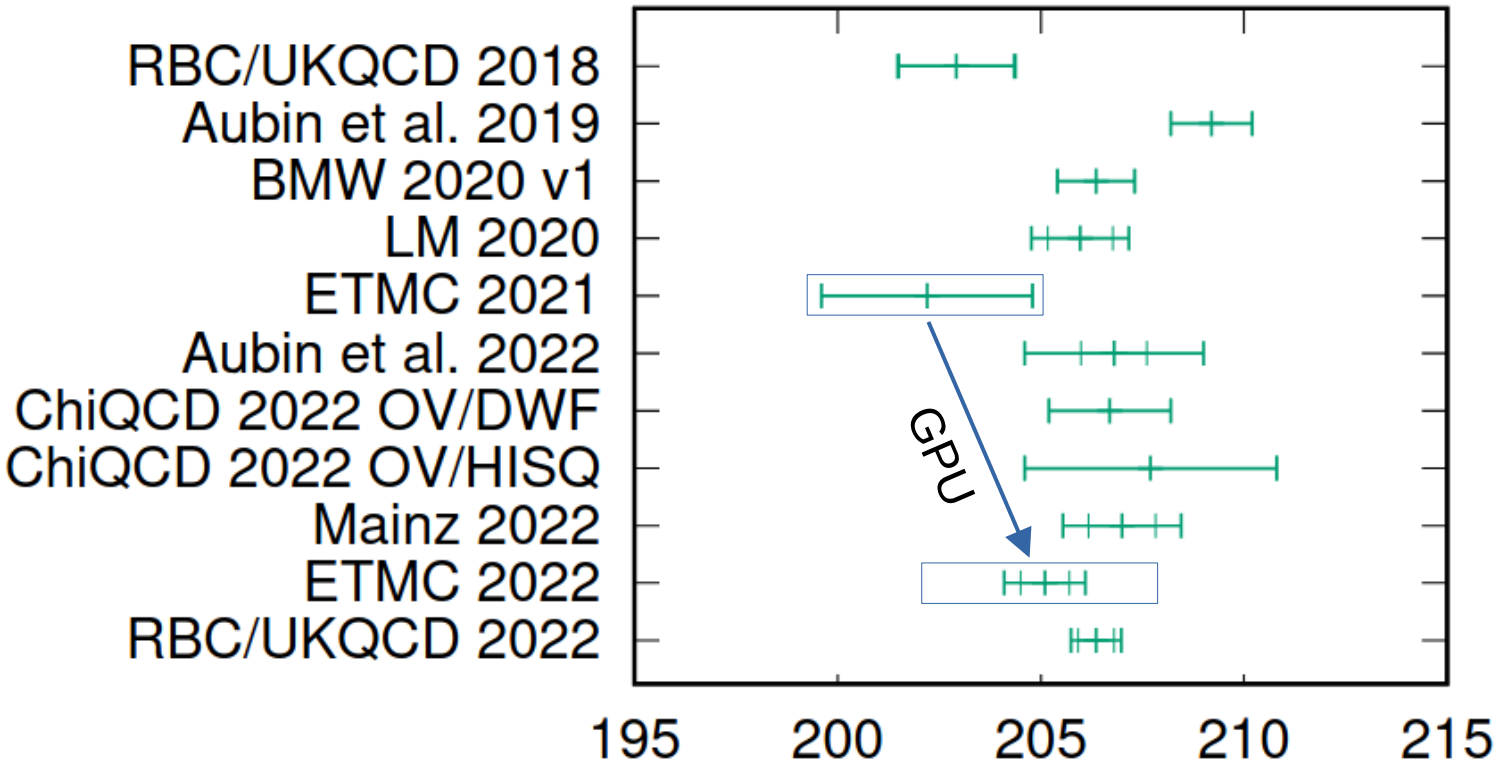
Numerical solution of the discrete Dirac Equation
(partial derivative equation → large sparse matrix)

A long list of scientific achievements: reconstruction of the hadron spectrum, thermodynamics of strong interactions, calculation of hadronic vacuum polarization...



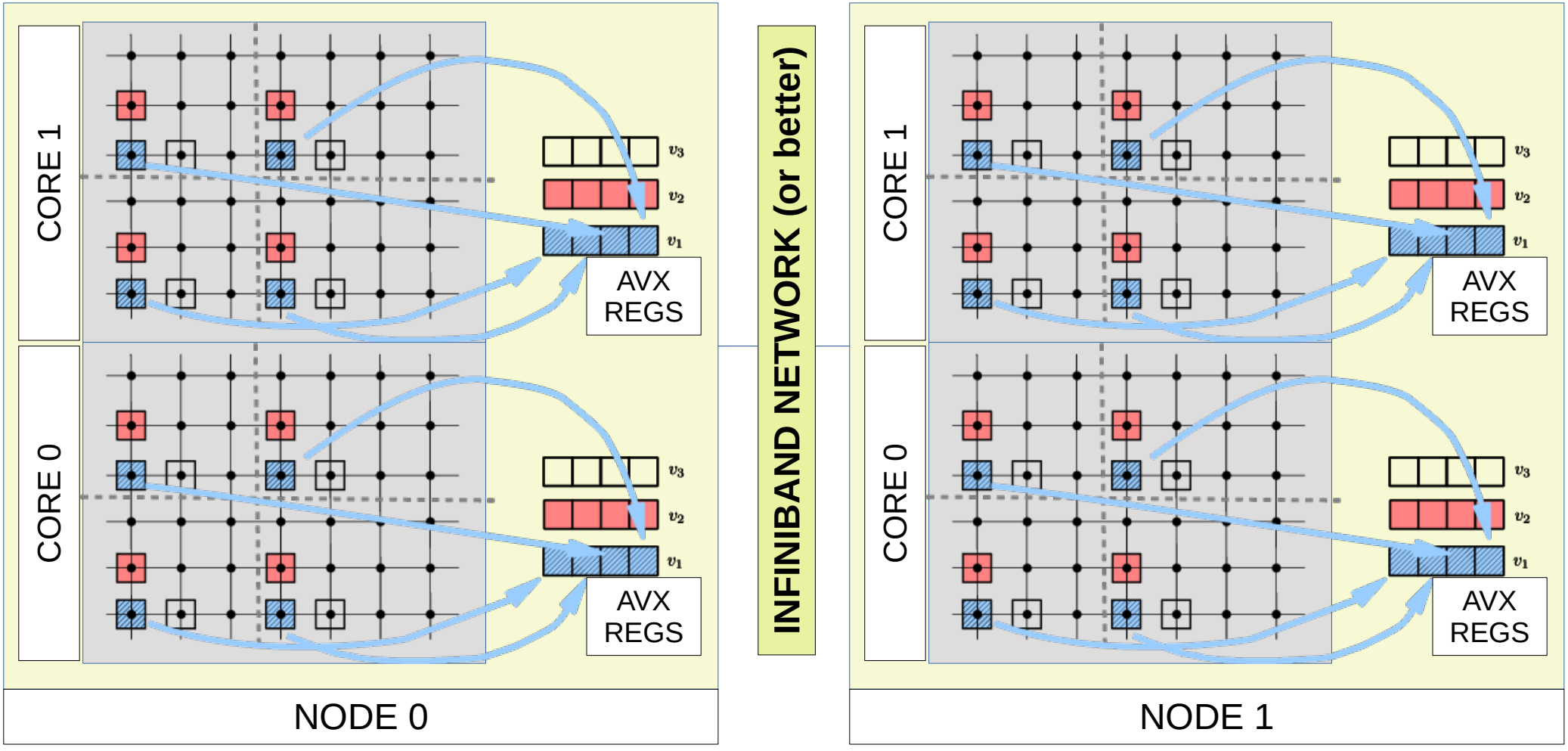
An example of a recent community effort where GPUs are making a difference





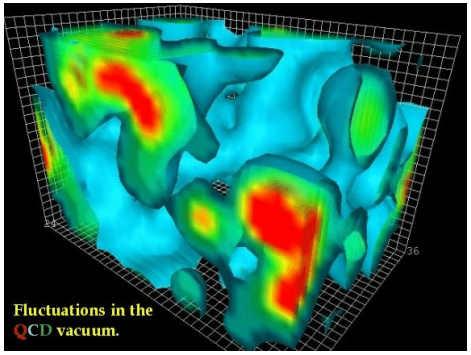
Hadronic vacuum polarization contribution to the muon anomalous magnetic moment (IW)

Massive parallelization scheme: **O(100) NODES** with **O(50) CORES** with **O(16) AVX REG**



Various strategies studied to adapt to GPU, most promising: AVX REGS → GPU THREADS

Typical lattice QCD simulation/measurement scheme



Producing O(100-1000) "**configurations**" of gluonic fields.

TOOL: Molecular Dynamics + Monte Carlo to evolve configurations of **gluonic fields,** the background in which quark particles "move".

RESOURCES: 1 configuration ~ O(1-50 GB data) ~ 1 day of simulation on **O(5000) cores**. Hundreds of MCorehours gained through national, European & worldwide supercomputing calls. Similar in spirit to the production of collisions at particle accelerators (tens of TB of data: WP5).

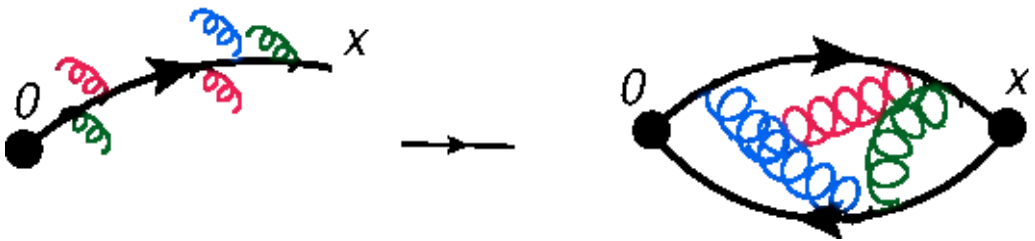
PORTING: Needs a number of different tasks to evaluate "**forces**" driving the dynamics, the most critical can be easily offloaded to GPU (used as accelerators) but **not all aspects** can be easily ported. A few fully ported frameworks are available (see next slides).

A **few large collaborations** ("big experiments") with important difference on the discretization. Multi-year "Runs", with statistics & systematics improving in time.

GOOD: cross checks, universality, healthy competition.
BAD: proliferation of codes.

Typical lattice QCD simulation/measurement scheme

Propagating O(100) quark on the gluon field backgrounds, take some algebraic combination:



TOOL: Numerical solution of Dirac Equation, tensor algebra to manipulate many spin and color degrees of freedom.

RESOURCES: 100 propagator ~ 1 hour of simulation on O(5000) cores.
Similar in spirit to data analysis of collision events. “Smaller” national, European calls.

PORTING: Several efficient numerical solvers for CPU & GPU, tensor algebra more tricky.

More collaborations of smaller scale with more specific problems & more code platforms.

GOOD: the critical task is the same for everybody, solved thanks to efficient libraries on GPU.
BAD: the remaining part of the code can still have a significant cost and is not homogeneous.

LATTICE QCD SIMULATIONS HARDWARE & SOFTWARE

A BRIEF HISTORY...

EARLY INVOLVEMENT IN BUILDING CUSTOM SUPERCOMPUTERS
(and custom programming language)

1986

THE APE COMPUTER: AN ARRAY PROCESSOR OPTIMIZED FOR LATTICE GAUGE THEORY SIMULATIONS

M. ALBANESE [d], P. BACILIERI [a], S. CABASINO [b], N. CABIBBO [c], F. COSTANTINI [d], G. FIORENTINI [d], F. FLORE [d], L. FONTI [a], A. FUCCI [e], M.P. LOMBARDO [d], S. GALEOTTI [d], P. GIACOMELLI [h], P. MARCHESINI [e], E. MARINARI [c], F. MARZANO [b], A. MIOTTO [f], P. PAOLUCCI [b], G. PARISI [c], D. PASCOLI [f], D. PASSUELLO [d], S. PETRARCA [b], F. RAPUANO [b], E. REMIDDI [a,g], R. RUSACK [h], G. SALINA [b] and R. TRIPICCIONE [d]

[a] *INFN-CNAF, Bologna, Italy*
[b] *Dipartimento di Fisica, I Universita' di Roma "La Sapienza" and INFN-Sez. di Roma, Italy*
[c] *Dipartimento di Fisica, II Universita' di Roma "Tor Vergata" and INFN-Sez. di Roma, Italy*
[d] *Dipartimento di Fisica, Universita' di Pisa and INFN-Sez. di Pisa, Italy*
[e] *CERN, Geneva, Switzerland*
[f] *Dipartimento di Fisica, Universita' di Padova and INFN-Sez. di Padova, Italy*
[g] *Dipartimento di Fisica, Universita' di Bologna and INFN-Sez. di Bologna, Italy*
[h] *The Rockefeller University, New York, USA*

The APE computer is a high performance processor designed to provide massive computational power for intrinsically parallel and homogeneous applications. APE is a linear array of processing elements and memory boards that execute in parallel in SIMD mode under the control of a CERN/SLAC 3081/E. Processing elements and memory boards are connected by a 'circular' switchnet. The hardware and software architecture of APE, as well as its implementation are discussed in this paper. Some physics results obtained in the simulation of lattice gauge theories are also presented.

LATTICE QCD COMMUNITY HAS BEEN LONG ACTIVE IN USING GPU

2007!!!!!

Lattice QCD as a video game

Győző I. Egri[a], Zoltán Fodor[abc], Christian Hoelbling[b],
Sándor D. Katz[ab], Dániel Nógrádi[b] and Kálmán K. Szabó[b]

Comput.Phys.Commun.177:631-639,2007

[a] *Institute for Theoretical Physics, Eötvös University, Budapest, Hungary*
[b] *Department of Physics, University of Wuppertal, Germany*
[c] *Department of Physics, University of California, San Diego, USA*

Abstract

The speed, bandwidth and cost characteristics of today's PC graphics cards make them an attractive target as general purpose computational platforms. High performance can be achieved also for lattice simulations but the actual implementation can be cumbersome. This paper outlines the architecture and programming model of modern graphics cards for the lattice practitioner with the goal of exploiting these chips for Monte Carlo simulations. Sample code is also given.

ITALIAN CODE FOR FULL HMC SIMULATIONS: C & CUDA, SINGLE GPU

2011

QCD simulations with staggered fermions on GPUs

C. Bonati[a], G. Cossu[b], M. D'Elia[c], P. Incardona[a]

[a] *Dipartimento di Fisica, Università di Pisa and INFN, Largo Pontecorvo 3, I-56127 Pisa, Italy*
[b] *KEK IPNS, Theory Center, 1-1 Oho, Tsukuba-shi, Ibaraki 305-0801, Japan*
[c] *Dipartimento di Fisica, Università di Genova and INFN, Via Dodecaneso 33, I-16146 Genova, Italy*

Abstract

We report on our implementation of the RHMC algorithm for the simulation of lattice QCD with two staggered flavors on Graphics Processing Units, using the NVIDIA CUDA programming language. The main feature of our code is that the GPU is not used just as an accelerator, but instead the whole Molecular Dynamics trajectory is performed on it. After pointing out the main bottlenecks and how to circumvent them, we discuss the obtained performances. We present some preliminary results regarding OpenCL and multiGPU extensions of our code and discuss future perspectives.

Keywords: Lattice QCD, Graphics Processing Units

PRO: already running on GPU more than 10 years ago!
CON: deep coded, CUDA used at the user level, single GPU, no portability...

PORTABILITY THROUGH OpenACC, LIMITED MULTIGPU SUPPORT

2018

International Journal of Modern Physics C | Vol. 29, No. 01, 1850010 (2018) | Research Papers

No Access

Portable multi-node LQCD Monte Carlo simulations using OpenACC

Claudio Bonati, Enrico Calore, Massimo D'Elia, Michele Mesiti, Francesco Negro, Francesco Sanfilippo, Sebastiano Fabio Schifano, Giorgio Silvi and Raffaele Tripiccione

https://doi.org/10.1142/S0129183118500109 | **Cited by:** 12

Abstract

This paper describes a state-of-the-art parallel Lattice QCD Monte Carlo code for staggered fermions, purposely designed to be portable across different computer architectures, including GPUs and commodity CPUs. Portability is achieved using the OpenACC parallel programming model, used to develop a code that can be compiled for several processor architectures. The paper focuses on parallelization on multiple computing nodes using OpenACC to manage parallelism within the node, and OpenMPI to manage parallelism among the nodes. We first discuss the available strategies to be adopted to maximize performances, we then describe selected relevant details of the code, and finally measure the level of performance and scaling-performance that we are able to achieve. The work focuses mainly on GPUs, which offer a significantly high level of performances for this application, but also compares with results measured on other processors.

PRO: portable through GPU/CPU thanks to extended C syntax
CON: limited multigpu parallelism, compiler specific, fixed (suboptimal) memory layout

MORE APPROACHES AND PARADIGMS

QUDA LIBRARY – M.Clark et al., since 2009

Heterogeneous collection of solvers for the Dirac equation, with a number of modern and adaptative algorithms, supporting various lattice QCD regularizations.

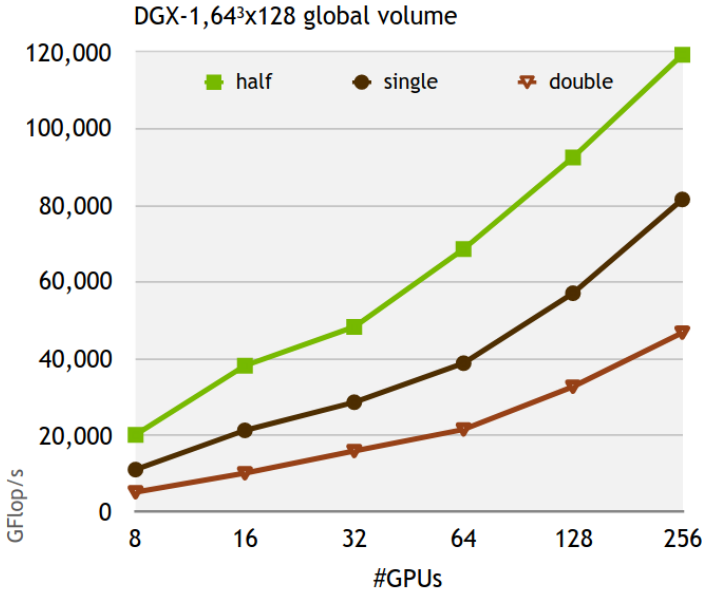
Open Source, actively developed by NVIDIA, through a strong group of former lattice QCD researcher. Makes use of all edge cutting GPU technology available.
Employed by several lattice QCD groups around the world.

PRO

Extremely well performing for the supported tasks

CON

Cannot perform all typical lattice QCD tasks (no full HMC).
Extremely difficult to adapt to different tasks from supported.
No portability.
(+ crazy interface & terrible documentation...)



GRID LIBRARY – P.Boyle et al. since 2015

C++ framework for the calculation of correlation functions & full HMC simulations (?)
Targetting a number of Lattice QCD regularization, easy to extend, efficient

Frontend: modern C++ 11 with a bit of metaprogramming + Python interface
Backend: supporting several archtectures: Cuda, HIP, OpenMP, etc (kernel abstraction)

PRO

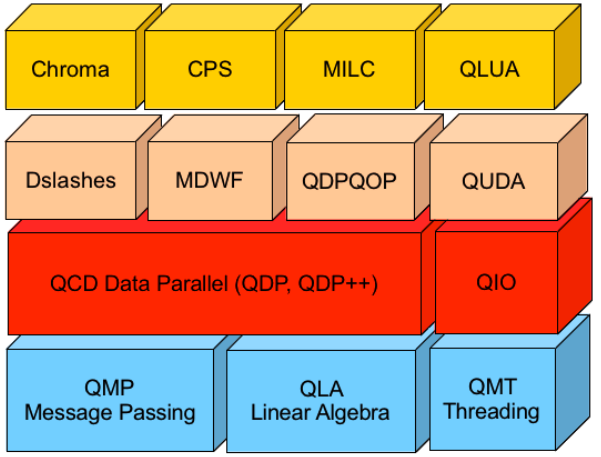
Intuitive, multiplatform, reasonably efficient on all platforms, relatively lightweight, adopting optimal memory layout transformations to efficiently use the resources.

CON

Reduced community (mostly US/UK oriented), limited expertise available in Italy
Engaging with the developers proved not easy in the past.

MILC software stack from USQCD software stack

Large software stack for HMC simulations & measurements.
Mainly used in the US & UK, a few users in Italy.



PRO

- Large community (in the US),
- Multiplatform.

CON

- Incomplete GPU support (multigpu?)
- Documented? Mhhh...
- Not trivial to setup (quite bloated code),
- Targeting a subset of the lattice interest.

NISSA LIBRARY – F.S. since 2011

In use from two major collaborations (LQCD123, PISA group)
Employed within several PRACE projects (PRA17-4394, PRA20-5171, PRA22-5171…)

Frontend: C++ 11 (envisaged migration to pure abstract C++17 metaprogrammed)
Backend: kernel abstraction, linked to several external libraries (including QUDA)

PRO

- “Large” user platform in Italy.
- Targeting different Lattice QCD regularization, multigpu & multithread.

CON

Missing the memory layout transformations to support more efficiently GPU & vector CPU for non-critical but important tasks

...AND MORE! (e.g. OpenQCD used in MiB)

SUMMARY OF THE STATUS

Historically: Long tradition of involvement in massively paralellization, with success on a number of different architecture including custom ones (since the ‘80s).

Typical architectures nowadays: Multithread AVX CPU and GPU.

Porting status: substantial parts of the calculation ported to GPU,

- Very good efficiency for critical tasks on specific architecture (NVIDIA),
- A few frameworks allows full abstraction on multigpu with good efficiency.

Still missing in the fully portable suits:

- several important use cases (e.g. observables in MILC),
- technical features integration for more efficient usability (e.g. memory layout NISSA),
- training of the users (e.g. GRID).

Future Strategy: selecting one (or more) suites & finalizing the implementation of the missing aspects. Choosing a single reference framework would ease work & enhance the impact.

Manpower? Open Calls?