The simulations in lattice QCD:

- are large scale and require massive parallelization,
- consume a lot of computing resources,
- and require constant algorithmic development and optimization for new hardware.

**openQCD Code**

openQCD simulation package consists of several programs for generating gauge configurations with \( O(a) \)-improved Wilson fermion action, allowing for several different choices of boundary conditions:

- open,
- periodic,
- Schrödinger Functional (SF), and
- open-SF.

This simulation program is based on the Hybrid Monte Carlo algorithm [3] and supports parallelization in 0, 1, 2, 3 or 4 dimensions. All the programs in this package are highly optimized for machines with current Intel or AMD processors, but will run correctly on any system that complies with the ISO C89 and the MPI 1.2 standards. The code is open source GPL, and is available for download from [http://luscher.web.cern.ch/luscher/openQCD/](http://luscher.web.cern.ch/luscher/openQCD/).

**Benchmarking: YM1**

We apply the first set of benchmarking on the program ym1 that generates an ensemble of gauge fields representative of the (pure) SU(3) gauge theory. Exactly which theory is simulated depends on the parameters passed to the program. For the tests in this presentation, we chose tree-level improved Symanzik gauge action.

**Scaling of openQCD**

- Strong scaling, global lattice: \( L = 32^4, \beta = 6.0 \)
- Weak scaling, local lattice: \( L = 8^4, \beta = 6.0 \)

Strong and weak scaling of the code on a typical Cray machine (PizDaint, CSCS).

**HPC Architectures**

<table>
<thead>
<tr>
<th>Machine</th>
<th>Platform</th>
<th>Software</th>
<th>Accelerator</th>
</tr>
</thead>
<tbody>
<tr>
<td>PizDaint, CSCS</td>
<td>Titan2</td>
<td>Intel Xeon</td>
<td>K-computer</td>
</tr>
<tr>
<td>Sunway Tofu</td>
<td>Titan2</td>
<td>AMD Opteron</td>
<td>cray/orca</td>
</tr>
<tr>
<td>TH Expert-1</td>
<td>Titan2</td>
<td>IBM POWER E6</td>
<td>cray/orca</td>
</tr>
</tbody>
</table>

**Initial Evaluation of openQCD**

We tested ym1 program in the case of 4 and 8 processors on the computing platform:

- 2 Dodeca-core Haswell Intel Xeon E5-2680

**Next Steps**

- Better communication modeling
- Increase number of processors
- Calibrate other supercomputers (Stampede, Blue Waters, MareNostrum, CSCS, etc.)

The goal is to reproduce the correct scaling on a single machine and devise a general method for the prediction of the scaling on an arbitrary machine.

**Future Machines**

The same procedure can be applied on machines that are still in the design phase, once the following features are known:

- machine topology
- processor speed
- communication characteristics

Nevertheless, it is hard to correctly model an unknown machine and the results must be interpreted carefully[4].

**Acknowledgements**

This work was supported by a grant from the Swiss National Supercomputing Centre (CSCS) under project ID 6424. Experiments presented in this paper were carried out using the PlaFRIM experimental testbed, being developed under the Inria PlaFRIM development action with support from LABRI and IMB and other entities: Conseil Régional d’Aquitaine, Université de Bordeaux and CNRS.

**References**

2. Casanova et al, [http://hal.inria.fr/hal-01017319](http://hal.inria.fr/hal-01017319)
4. Luca Stanisic, PhD Thesis. University Grenoble Alpes, 2015 [https://hal.inria.fr/tel-01248109v2](https://hal.inria.fr/tel-01248109v2)