PROCEEDINGS OF SCIENCE

Analysis and Visualization Tools for Lattice QCD

Massimo Di Pierro[*]
School of Computing - DePaul University - Chicago, IL - USA
E-mail: `mdipierro@cs.depaul.edu`

Yaoqian Zhong
School of Computing - DePaul University - Chicago, IL - USA
E-mail: `ati_zhong@hotmail.com`

We developed a toolkit for analysis and visualization of QCD data. The analysis tools include a web based application for extracting data in real time from log files, computing autocorrelation, moving averages, bootstrap samples and bootstrap errors from user provided functions, non-linear Bayesian correlated fits, make the relative plots, and facilitate collaborative work. The visualization tools include standard QCD algorithms with the additional ability to save intermediate steps of the computation in parallel in the VTK file format for real time visualization of on-going computations. To demonstrate, we made visualizations showing the convergence of the MinRes inverter for different sources, evolution of the topological charge under different algorithms for dynamical fermions, energy density in space in the presence of static quark-antiquark, and heavy-light wave functions. Our toolkit is self contained and works in cooperation with many existing QCD code.

The XXVII International Symposium on Lattice Field Theory - LAT2009
July 26-31 2009
Peking University, Beijing, China

[*] Speaker.

1. Introduction

Our work consists of building computer programs to facilitate collaboration, analysis and presentation of data for Lattice QCD physicists. We mainly distinguish two classes of such tools:

- A web based application (also available as a self-standing GUI tool) for collaboration, Monte Carlo analysis and 2D visualization of analysis results.

- A collection of parallel algorithms for generating 3D images and animations from Lattice QCD gauge configurations.

We briefly describe each of these tools in the following two sections.

2. Analysis and 2D Visualization Tools

In a typical Monte Carlo computation [1] one can distinguish two main phases. In Phase I, a computationally intensive Markov Chain Monte Carlo (MCMC) program generates gauge configurations and, for each configuration, measures multiple correlation functions. In Phase II, the output of the above program is analyzed to produce averages and plots. While Phase II requires less computation than I, it is not less expensive in human time, and the one that requires the most collaboration among people. For this reason we have created a web based tool called *mc4qcd* that performs the following steps:

- It **acquires** data from the users, in the form of log files from phase I. The system does not dictate a file format for these log files. It just requires that they contain multiple measurements for the same quantity (once per gauge configuration) and that each quantity is labeled using a prefix. The prefix may contain an index. For example, the log file may contain lines similar to “2pt[4] = 0.121235” or “2-point correlation function at 4: 0.121235”. The log file can contain additional text that will be ignored in the parsing.

- It **parses** the data using regular expressions to extract quantitative information. The regular expressions are provided by the user, can be different for each log file, and may be applied to multiple runs. An example could be: “Extract all 2pt and 3pt correlation functions.”

- It **filters** the data based on user preferences. For example “consider only 2pt correlation functions separated by less than 12 time-slices”.

- It **mines** the data for expressions of interest. For example “compute all possible ratios of $3pt[<t_1>][<t_2>]/(2pt[<t_1>]\times 2pt[<t_2>])$”.

- It **represents** the data in multiple formats such as text based, plots and histograms. For example “show an histogram of the bootstrap samples for $3pt/2pt^2$”. The various plot types include: raw data, autocorrelations, moving averages, bootstrap samples, averages with bootstrap errors.

- It allows one to further **refine** the data and **interact** with the data by fitting the bootstrap results using Bayesian correlated fits with arbitrary non-linear functions.

The entire process is executed via a browser, it requires login, and enforces access control on all data at every step. This facilitates collaboration between members of a group by allowing them to easily locate and search data and plots. Users can also comment on each other's data (if they have access to it) and the comments can include Latex expressions. In this way, the thought process of the physicist working on the data is stored together with the data in the web application. The underlying libraries can be used to batch script all the functions provided by the web application.

Figure 1 shows some screenshots from the *mc4qcd* program. The top-left screenshot lists the data sets available to a user, links possible analysis algorithms, and analysis results. The plots represent an example of autocorrelation functions for the $2pt$ at 3 different time slices, the histograms of the bootstrap samples, and a plot of $2pt[t]$ with an exponential fit as function of t.

Although this program was designed specifically for lattice QCD in mind, it can be used for the analysis and plotting of MCMC data in other areas of Physics.

The program is packaged together with a web server and a file based relational database into a single application that runs on Windows, Mac and Linux. Its only dependencies are the Python interpreter and the matplotlib library.

3. 3D Visualization Tools

Typical Lattice QCD computations generate a large quantity of data that is normally only looked at in aggregated form (numbers with a precision of a few digits and simple 2D plots). A lot of the information is discarded. While this is the nature of MCMC computations, we believe there may be a value in looking in more detail at the data being generated in order to identify patterns that may lead to better understanding of the algorithms and/or that may be symptomatic of errors in the computations. Moreover, it will better present scientific results and educate the public about Lattice QCD.

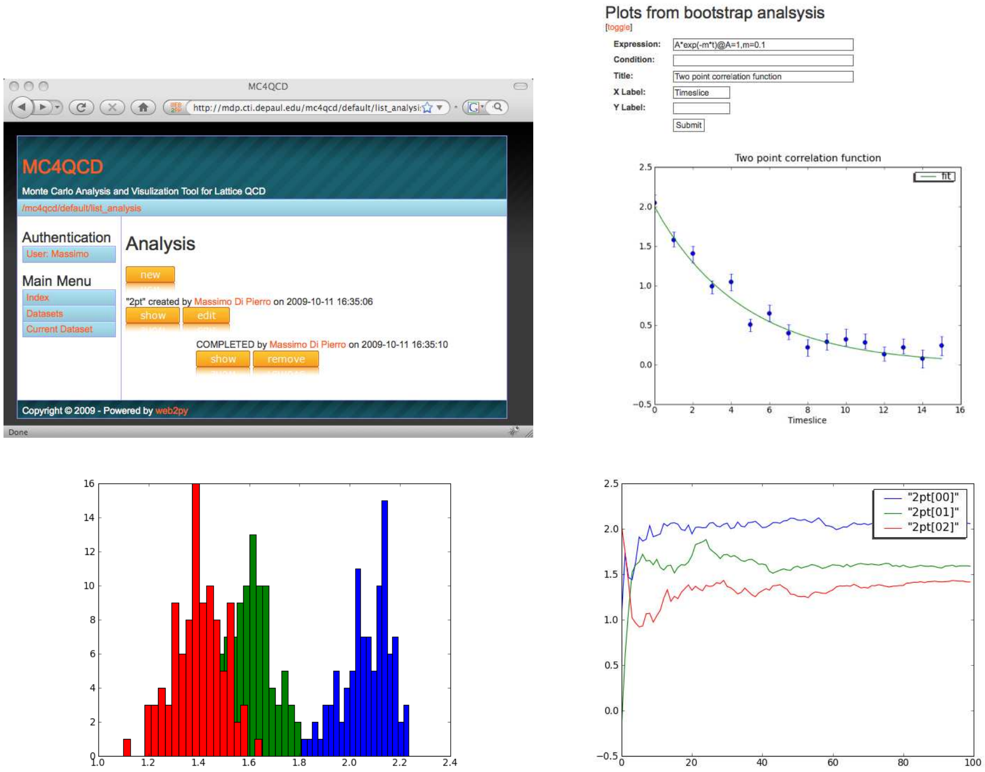
Our toolkit comprises of a set of Lattice QCD libraries for implementing actual computations. These libraries have been modified to be able to save their intermediate state, in parallel, into fields (gauge, fermions, scalars) which can be visualized in 3D interactively (volume plots, iso-surfaces, slices).

Our toolkit can read multiple gauge configurations formats including ILDG, Lime, Nersc (3x2 and 3x3), MILC and FermiQCD [2]. The file format is automatically detected and each file is converted into a FermiQCD format for further analysis.

The output fields (for example the energy density, the topological charge, the component of a propagator) are saved as scalars and VTK files. Each time-slice is saved as a separate scalar field within the same VTK file. The VTK files can be combined using an mpeg-encoder to make movies.

The Visualization Toolkit (VTK) is an open source, platform independent, graphics engine with libraries for parallel rendering of 3D visualizations. VTK is being developed by a large collaboration of universities, laboratories (Sandia, Los Alamos, and Lawrence Livermore), and private companies. VTK provides core functionality for many existing visualization packages such as VisIT (developed by LLNL), MayaVi and Paraview. All these packages are compatible and can interoperate with our software.

Here we present two case studies of usage of our toolkit.



MC4QCD

http://mdp.cti.depaul.edu/mc4qcd/default/list_analysi

MC4QCD

Monte Carlo Analysis and Visulization Tool for Lattice QCD

/mc4qcd/default/list_analysis

Authentication

User: Massimo

Main Menu

Index

Datasets

Current Dataset

Analysis

new

"2pt" created by Massimo Di Pierro on 2009-10-11 16:35:06

show edit

COMPLETED by Massimo Di Pierro on 2009-10-11 16:35:10

show remove

Copyright © 2009 - Powered by web2py

Done

Plots from bootstrap analsysis

[toggle]

Expression: A*exp(-m*t)@A=1,m=0.1

Condition:

Title: Two point correlation function

X Label: Timeslice

Y Label:

Submit

Figure 1: The top-left image shows a page of the mc4qcd program. This page allows a user to select a data-set and run one or more analysis algorithms on that data-set. The top-right image shows a typical plot generated by the system for a 2 point correlation function analysis on the database. The images at the bottom show, from left to right respectively, the distribution of bootstrap samples and moving averages for $2pt[t]$ at three different values of t. These plots and more are generated automatically by the web application upon upload of a data-set.

3.1 Autocorrelation length

In this case study [3] we visualize topological charge density and measure how fast or slow it evolves under the Markov Chain Monte Carlo (MCMC) for different algorithms and different simulation parameters (the dynamic quark mass in particular). This would give us strong confidence that current Lattice QCD computation are adequately sampling the space of all possible configurations as opposed to getting stuck in one topological sector, and provide an effective tool for developing algorithms that maximize equilibration of the topological charge. We analyzed ad hoc gauge configurations generated with small trajectories by Michael Clark and Chulwoo Jung respectively:

- Two ensembles of 32×24^3 dynamical Wilson gauge configurations with $\beta = 5.6$ and quark masses of 66MeV and 33MeV respectively. ($dt = 0.1$)
- One ensemble of 64×24^3 dynamical Domain Wall gauge configurations with beta=2.13, two degenerate light quarks of 14MeV, and one strange quark of 74MeV. ($dt = \frac{1}{6}$).

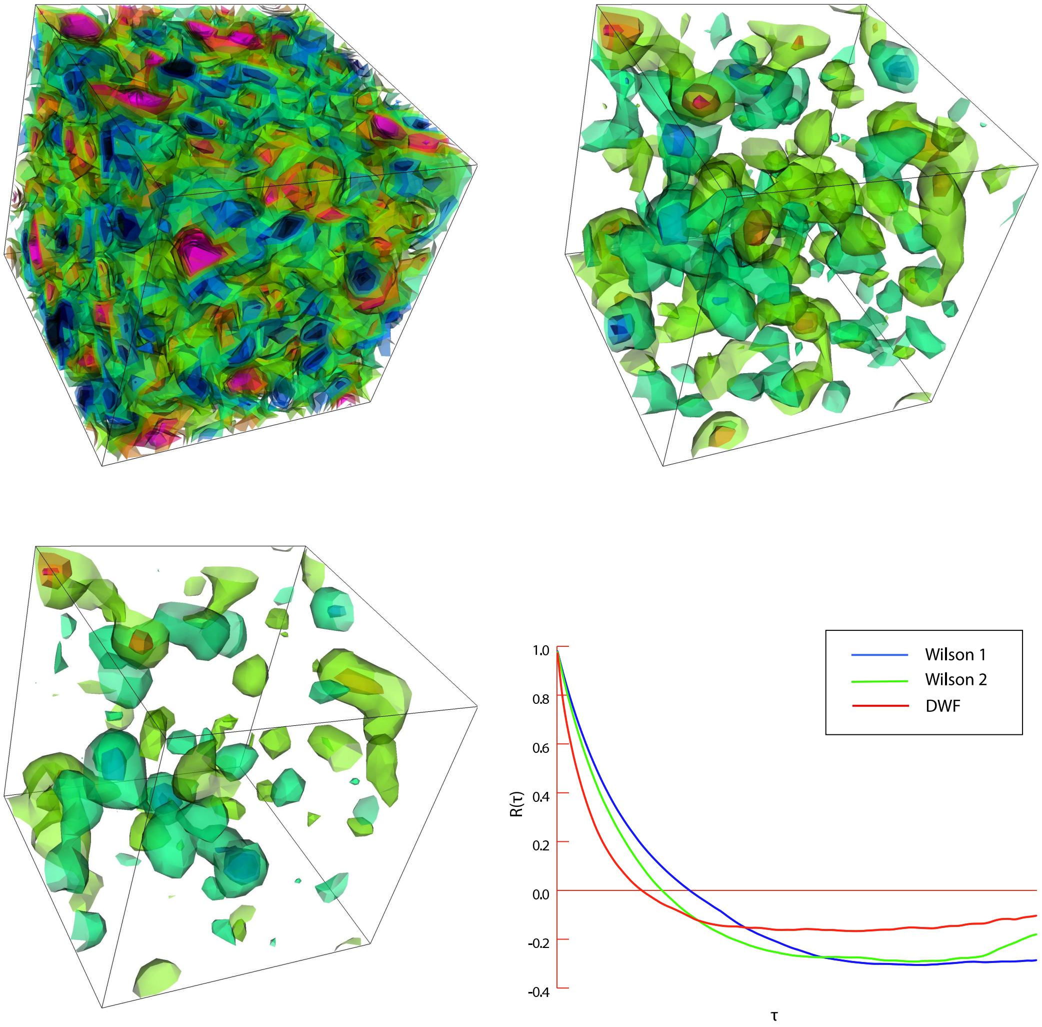


Figure 2: The images show topological charge density at different cooling steps for a sample gauge configuration. The plot at the bottom-right shows the autocorrelation length for the computed topological change density for the three ensembles of production quality gauge configurations discussed in the paper.

We performed 20 gauge invariant APE smearing steps for cooling, shown in figure 2, computed the topological change, and computed the average autocorrelation of the topological change density.

The result is shown in bottom-right plot. The autocorrelation crosses 0 after about 100 steps. This indicates that the local topological charge becomes de-correlated after about 10 MCMC steps at $dt = 1$ which is typical for production grade Lattice QCD computations.

3.1.1 Inversion Residue

Visualization can be used to represent various types of fields; for example, wave functions, form factors, and/or bare quark propagators. Visualizing the latter can be useful to understand the rate of convergence of the inverters at a local level as opposed to at a global level (as measured by the total residue). In this example we show how we have modified the MinRes inverter to visualize a quark propagator on a typical production quality gauge configuration:

```
class MinResVtk {
public: static inversion_stats inverter(...) {
    ...
    psi_in.update();
    mul_Q(r,psi_in,U,coeff);
    r*=-1;
    r+=psi_in;
    psi_out=psi_in;
    do {
      r.update(); // sync parallel buffers
      mul_Q(q,r,U,coeff);
      alpha=q*r;
      alpha/=norm_square(q);
      mdp_add_scaled_field(psi_out, alpha, r);
      mdp_add_scaled_field(r, -alpha, q);
      residue=sqrt(norm_square(r)/r.global_size());
      forallsites(x) // project into a scalar
        for(int a=0, s(x)=0; a<4; a++)
          for(int k=0; k<psi_in.nc; k++)
            s(x)+=sqrt(real(psi_out(x,a,k)*conj(psi_out(x,a,k))));
      s.save_vtk(filename,...); // save as VTK file
    } while (...);
    return stats;
  }
}
```

Figure 3 show the absolute value of a quark propagator for a planar source (top) and a solid source (bottom), after a few inversion steps (left) and after convergence (right), for the same gauge configuration. The visualization indicates that the inversion algorithm converses uniformly everywhere.

4. Conclusions

We believe visualization will prove to be an important tool for understanding Lattice QCD algorithms and the MCMC evolution. For example the exact nature of the topological objects responsible for confinement is yet to be understood satisfactorily. We also believe visualization can play an important role in education and outreach outside the core of the Lattice QCD community. Our tools are Open Source (GPL2) and can be downloaded from:

```
https://launchpad.net/qcdmc
https://launchpad.net/qcd
```

Acknowledgements

We thank Argonne National Laboratory for proving computing resources Michael Clark (BU) and Chulwoo Jung (BNL) for proving the USQCD gauge configurations used in some of our im-

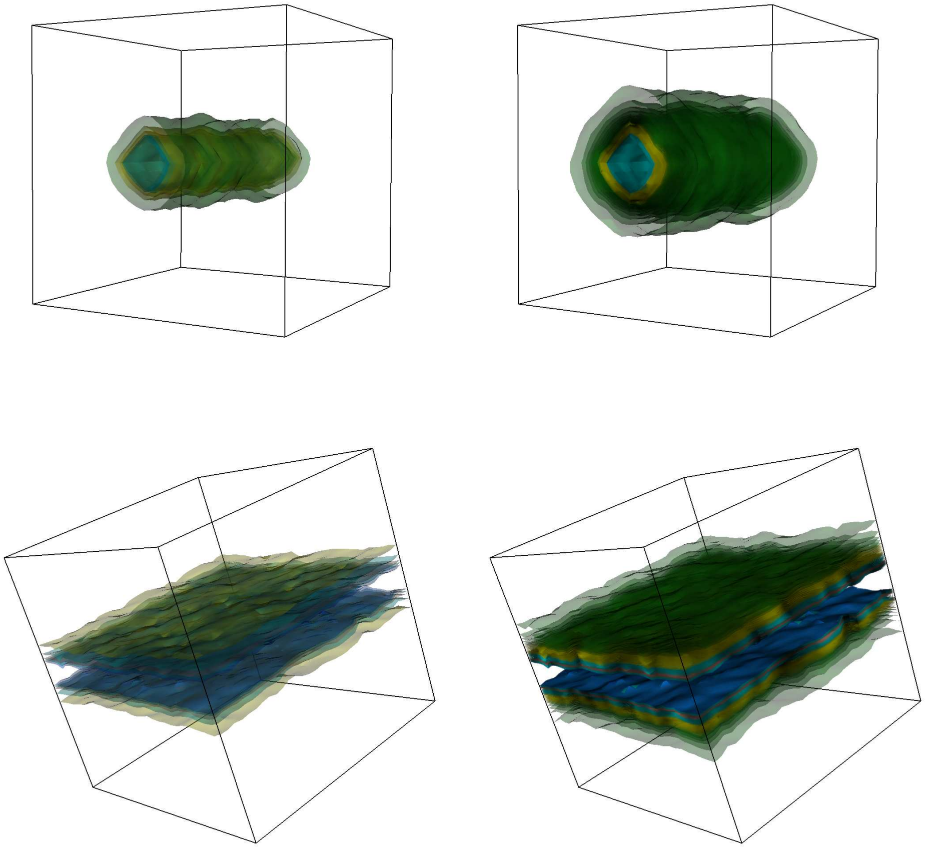


Figure 3: The first three images show the topological charge for a typical gauge configuration at different cooling steps. The plot at the bottom-right shows auto-correlation for the three different ensembles of gauge configurations discussed in the paper.

ages. We also thank James Osborn (ANL), John Negele (MIT), Richard Brower (BU), and Steven Gottlieb (Indiana) for various comments and advice.

This project is funded by the US Department of Energy under grant DEFC02-06ER41441.

References

[1] M. Di Pierro, J. Mod. Physics A21 (2006)

[2] M. Di Pierro *et al.*, Lattice 2003, Nucl. Phys. B Proc. Suppl. (2004)

[3] M. Di Pierro *et al.*, J. Phys. Conf. Ser. 180 (2009) 012068