Visualization for Lattice QCD

Massimo Di Pierro

School of Computer Science, Telecommunications and Information Systems
DePaul University, 243 S Wabash Ave, Chicago, IL 60604, USA
E-mail: `mdipierro@cs.depaul.edu`

We present a prototype visualization toolkit for Lattice Quantum Chromodynamics. The toolkit consists of a set of parallel algorithms for the computation of the topological charge, energy density, density of heat-bath hits, two and three point correlation functions, that are interfaced with a Graphical User Interface for an interactive visualization of the fields. The toolkit allows both real-time and off-line visualization, scripting for automation, and the ability to combine individual frames into animations (to produce, for example, an animation of the topological charge as function of the MCMC step). The toolkit also includes analysis and plotting tools to automate the typical workflow of Lattice QCD computations.

The XXV International Symposium on Lattice Field Theory
July 30-4 August 2007
Regensburg, Germany

1. Introduction

Typical Lattice QCD simulations generate Tera-bytes of data in the intermediate steps of the computation but the results of each simulation consist of few numbers with a couple of digits of precision. We believe that the ability to visualize the data produced in the intermediate steps of the computations can provide additional value to current and future simulations. On the one hand, data visualization can facilitate the communication of physical concepts and strengthen the outreach effort of the lattice community. On the other hand, it will improve our ability to identify bugs in the algorithms and potential sources of bias in the computations, it will help find pathologies in the convergence properties of the inverters and, perhaps, improve our understanding of Lattice QCD.

Other authors in the past have generated excellent visualizations from Lattice QCD computations [1, 2]. Our project differs from those since it aims to build a comprehensive set of user friendly tools, as opposed to making specific visualizations.

Our toolkit consists mainly of a set of parallel algorithms for Lattice QCD that read typical data files, such as gauge configurations and propagators, and produce various types of projections into 4D gauge invariant scalar fields, for example the topological charge density. The 4D scalar fields are partitioned by time slice and stored together and in parallel in binary VTK files which can be read by most visualization applications including MayaVi, Paraview, VisIt and OpenDX. We have modified the MayaVi2 interface to allow real-time image update and animations. Visualizations and animations can be scripted in Python.

The toolkit also includes an analysis and plotting tool to extract information form log files and to study, both off-line and in real-time, the results of the computation (plot partial averages, sample distributions, moving averages, bootstrap errors, etc.).

Our toolkit is a work in progress and the long term goal is that of developing a comprehensive set of components for automating the workflow of Lattice QCD computations and extract visual information from them.

Many of the images shown in this paper, although they are the result of real Lattice QCD computations, should be considered qualitative in nature because their purpose is to explain the capabilities of the toolkit and not present new physics results. For this reason they have been computed on small lattices, relatively to modern state of the art computations; moreover the parameters (quark masses and gauge coupling) have not been tuned to match the physical ones.

The current version of the toolkit, additional images and some movies can be downloaded form `http://mdp.cti.depaul.edu/vqcd`.

2. Visualization Tools

The main component of our visuation toolkit is a library of parallel lattice QCD algorithms. Programs using the library are generated using meta-programming techniques and they do not require programming by the user. These programs can read gauge configurations generated by FermiQCD, UKQCD, MILC, Chroma and QDP++ code and apply algorithms in the desired order. For example 1) perform cooling, 2) compute the topological charge; 3) slice and save as VTK file.

VTK, or Visualization Tool Kit, is a standard format for storing large 3D data sets. It supports both structured and unstructured meshes, scalar, vectors and tensor fields. We designed a VTK
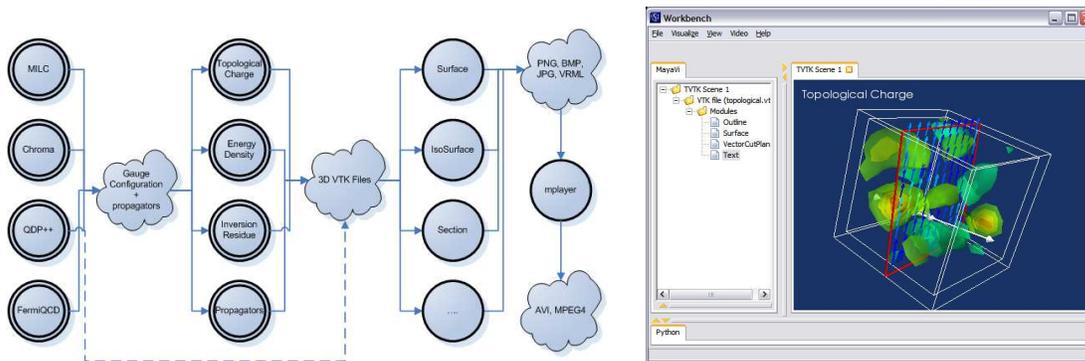
Figure 1: The figure represents the workflow diagram for our toolkit. The figure on the right is a screenshot of the modified MayaVi2 workbench. Via the workbench it is possible to interact with the visualization in real-time.

based file format to store the most general type of 4-dimensional field. All local field components are stored together and different time-slices are separated but stored in the same file.

In this way it is possible to use any standard visualization application to read the field and use the GUI to browse through the time-slices and the field components interactively. Most visualization applications have a pipeline design that allows to send the data through various filters, smoothing and rendering algorithms. The most common ways to render a 3d scalar field is by means of volume density plots or by iso-surfaces (fig. 1-left).

Different fields and algorithms can then be overlapped, rotated, scaled, projected and sliced interactively. For our toolkit we choose to use the MayaVi2 Workbench (fig. 1-right). This is an open source application written in Python that interacts with VTK via the MayaVi2 API. We modified the workbench and included the ability to loop over multiple LQCD files in order to handle large numbers of files at once (for example loop over all files in a folder, produce one image for each file and then encode the frames into an MPEG animation) and to monitor files for changes (for example to see how a field evolves while a LQCD computation is in progress).

In this paper we reproduce, as examples, screenshots of animations generated by our toolkit.

Fig. 2 shows isosurfaces for the topological charge (instantons) at different steps of the heath-bath Markov Chain. The initial gauge configuration is characterized by half of the lattice in a cold state and half in a hot state. A qualitative but interesting result of this visualization is that instantons do not appear to diffuse from the hot to the cold part of the lattice, instead their density(size) decreases(increases) in the hot half of the lattice and increases(decreases) in the cold half. We adopted the following definition of local topological charge in accord with ref. [3]

$$Q(x) = \text{Re} \left\langle F_{\mu\nu}(x)\tilde{F}_{\mu\nu}(x) \right\rangle \tag{2.1}$$

$F_{\mu\nu}$ is the smeared chromo-elector-magnetic field.

Fig. 3 shows the energy density for a $SU(2)$ gauge field in presence of a static quark and a static anti-quark as more gauge configurations are averaged. The initial noise (in the first image) dissipates and coherence causes the emergence of the flux tube connecting the quark and the anti-quark (in the third image). Our computation is performed without abelian projection using 1000
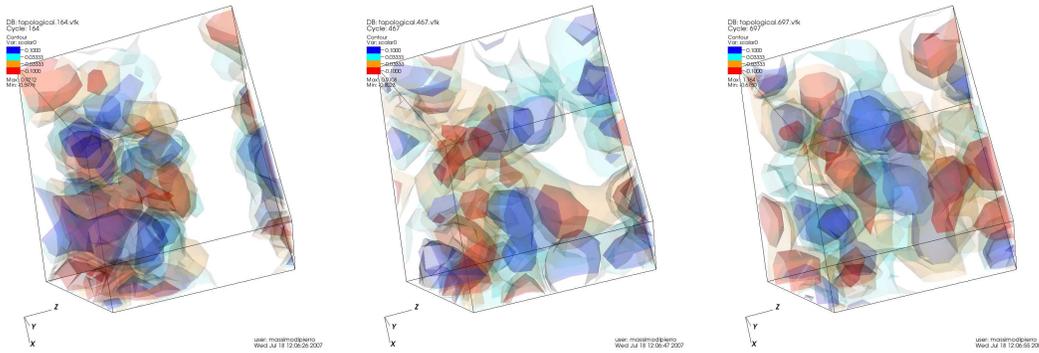
Figure 2: The figures show isosurfaces for the topological charge (instantons) at different steps of the heathbath Markov Chain. The initial gauge configuration is characterized by half of the lattice in a cold state and half in a hot state. A qualitative but interesting result of this visualization is that instantons do not diffuse from the hot to the cold part of the lattice, instead their denity(size) decreases(increases) in the hot half of the lattice and increases(decreases) in the cold half.
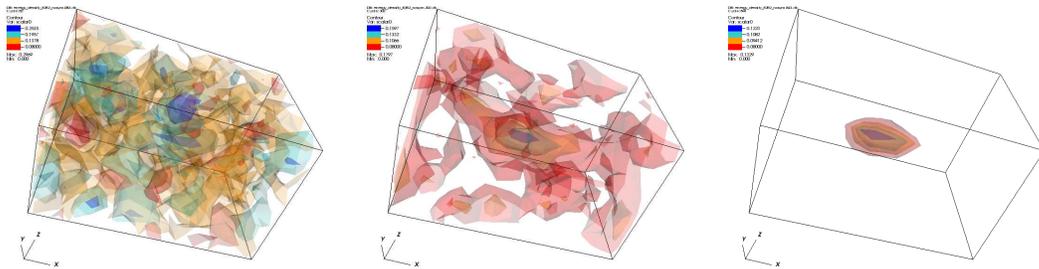

Figure 3: The figures show the energy density for a $SU(2)$ gauge field in presence of a static quark and a static anti-quark as more gauge configurations are averaged. The initial noise (in the first image) dissipates and coherence causes the appearence of the a flux tube connecting the quark and the anti-quark (in the third image).

gauge configurations and the result is qualitatively different from those computations with abelian projection. In fact, our case the peak in the energy density is at the center of the flux tube, while in the abelian projected case there are two peaks where the quarks are localized. We adopted the following definition of energy density, in accord with ref. [4]

$$V_{\mu\nu}(\vec{x}) = \frac{\langle WP_{\mu\nu}(T/2,\vec{x})\rangle}{\langle W\rangle} - \langle P_{\mu\nu}(T/2,\vec{x})\rangle \tag{2.2}$$

$V_{0i} \equiv E_i^2$, $V_{ij} \equiv \varepsilon_{ijk}B_k^2$ and $x = (T/2,\vec{x})$. P is a $SU(2)$ plaquette. W is the Wilson loop corresponding the static quark and anti-quark.

Fig. 4 shows the only non-zero components of a cold Wilson propagator, respectively ψ_{00}, ψ_{20} and ψ_{30}, i.e. solution of the Dirac equation $(\not{D} - m)\psi = \delta$ in absence of background field. δ is a delta function localized at the center of the lattice where only spin component 0 and color component 0 are different from zero.
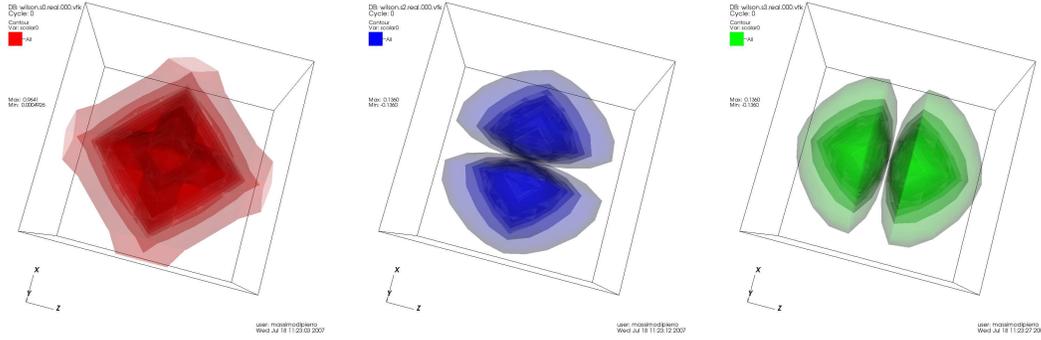
Figure 4: The only non-zero components of a cold Wilson propagator, respectively the real parts of ψ_{00}, ψ_{20} and ψ_{30}, i.e. solution of the Dirac equation $(\not{D} - m)\psi = \delta$ in absence of background field.
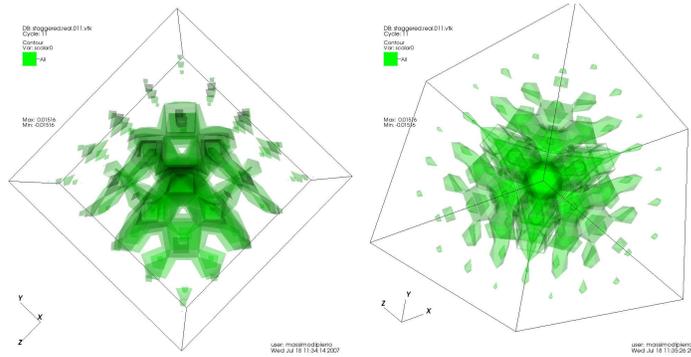

Figure 5: A cold staggered propagator, viewed from a face and viewed from corner of the lattice respectively..

Fig. 5 shows a cold staggered propagator, viewed from a face and viewed from corner of the lattice respectively. Because of symmetries of the action, the view is identical from and face and any corner.

Fig. 6 shows symmetric and anti-symmetric components of the light (valence) quark in a heavy-light meson. The resulting wave function is the QCD analogous of the chemical orbital for the hydrogen atom. They are computed by inserting a $\bar{q}q$ operator, at different locations in space, in between creation and annihilation operators for a B meson

$$f_\alpha(\vec{x}) = \langle B(+t) | \bar{q}_\alpha(\vec{x}) q_\alpha(\vec{x}) | B(-t) \rangle \tag{2.3}$$

Here $B(t) = \bar{h}(t,\vec{0})\gamma^5 q(t,\vec{0})$. h is a static quark.

3. Analysis Tools

Some of the critical components of many Lattice QCD computations are the Monte Carlo average, the study of convergence of the average, the estimation of the bootstrap error and analysis of the distribution of bootstrap samples. This process is repeated for many quantities, then final
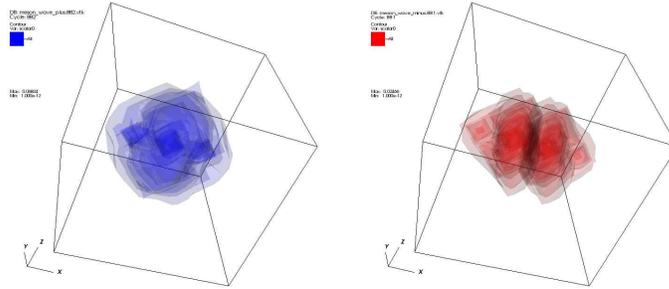
Figure 6: Symmetric and anti-symmetric components of the light quark in a B meson.

results are aggregated, fitted, and various types of plots are produced. Our analysis tool automates this workflow.

First, our system is provided with log (output) files containing various quantities measured on the individual gauge configurations. As an example, we consider log files that look like this:

```
2pt[0]=1.01232634
2pt[1]=0.33123125
...
3pt[0][0]=4.23452345
3pt[0][1]=5.52348454
...
```

for various combinations of the indices and for each gauge configurations. (The log files can also contain additional text, such as comments, that are of no interest here.)

Second, our system is provided with an expression like[1]

```
"3pt[<t1>][<t2>]"/("2pt[<t1>]"*"2pt[<t2>]")
```

The sub-expressions in quotes are the patterns to look for in the log files, the variables between <...> are defined by the expression itself and are macthed with the patterns. The rest is just a mathematical formula, function of the sub-expressions in quotes.)

Third, by clicking one button, the system does everything else:

- Matches all occurrences of "2pt[...]" and "3pt[...]" in the logfiles as function of $t1$ and $t2$.

- For each occurrence of "2pt[...]" and "3pt[...]" it computes autocorrelations, distributions, moving averages, and generates the corresponding plots.

- It builds all possible instances of the above expression, for example $3pt[3][4]/(2pt[3] * 2pt[4])$. For each of them it computes moving average, mean, bootstrap error, distribution of bootstrap samples, and generates the corresponding plots (including bar plots as function of $t1$ and $t2$.

---

[1]this is a typical operator that appers, for example, when computing the expectation value of an operator sandwiched between two creation/annihilation operators, as in eq.2.3.

6

- All plots (typically hundreds or thousands of them for each expression) and numerical results can be browsed via a Graphical User Interface

- Plots are interactive. They can be zoomed over and customized for printing.

- All bar plots allow fitting and extrapolations, including correlated, constrained, non-linear and bayesian fits using an easy to understand and natural syntax.

4. Conlcusions and Outlook

In this paper we have presented the general design and the current capability of a visualization tookit for Lattice QCD. It comprises of two parts: a set of parallel algorithms for computing quantities of physical interest and generating corresponding VTK files, and an analysis and plotting tool to automate the workflow of typical computations.

Although all the components shown here are fully functional, this project is still in its infancy. Our aim is to extend its functionality by adding components that may be of interest to physicists. Moreover the toolkit is in the process of being integrated with the broader USQCD software effort.

Acknowledgements

References

[1] D. Leinweber, Proceedings of "Workshop on Light-Cone QCD and Nonperturbative Hadron Physics", Adelaide, Australia, 13-22 Dec 1999. [hep-lat/0004025]

[2] M. Feurstein *et al.* Nucl. Phys. B (Proc. Supp.) Vol. 53, 1-3 (1997) pp.553-556

[3] F. Bonnet *et al.* Phys. Rev. D 62, 094509 (2000)

[4] D. G. Caldi and T. Sterling, Phys. Rev. Lett. 60, 24 (1988) pp.2454