DBW2 KPI

Jiqun Tu

1 Introduction

This is a manual of the package for measuring topological charge diffusion in the way described in [1]. We assume the potential users have basic understanding of `bash` system and lattice QCD calculation.

2 Package

The package is available at `git@github.com:hummingtree/DBW2-KPI.git`. In order to run wilson flow and measure topological charge `CPS`(Columbia Physics System) is needed: `git@github.com:hummingtree/cps-public.git`.

3 Install

3.1 CPS

We will need `gcc` and `MPI` to compile `CPS`. Make sure your `$PATH` includes `MPI` before compling.

1. Clone `CPS` from the git repository. Through out this document we will call this directory `$cps_repo`.
2. Go to the top of the repository, `$cps_repo`. `CPS` will be installed and placed under a directory specified as a variable `$cps`(line 4) in `$cps_repo/conf.sh` so change that if you want. The default directory is `../cps-build/public/`. Through out this document we will call this directory `$cps_build`.
3. Execute the `build.sh` script.

3.2 Analysis

We will need `g++`, `python`(version `2.7.5` tested) and `matlab`(version `R2013a (8.1.0.604)64-bit (glnxa64)` tested), later versions should be fine) to perform the whole analysis. Make sure these softwares are included in your `$PATH`.

1. Clone `DBW2-KPI` from the git repository. Through out this document we will call this directory `$DBW2-KPI_repo`.

2. Copy the file `$cps_build/build-qmp/Makefile.users` to both `$DBW2-KPI_repo/ens_gen/` and `$DBW2-KPI_repo/wflow_tcharge_cps/` as `makefile`, i.e.

```
$ cp $cps_build/build-qmp/Makefile.users $DBW2-KPI_repo/ens_gen/makefile
$ cp $cps_build/build-qmp/Makefile.users $DBW2-KPI_repo/wflow_tcharge_cps/
    makefile
```

4 Usage

There are two stages when using this package. In the first stage `CPS` reads in the lattice configurations in standard *nersc* format, run wilson flow and measure the topological charge density. The second stage process the raw data produced by `CPS` and perform the fit.

4.1 First Stage

1. Go to `$DBW2-KPI_repo/wflow_tcharge_cps/binaries`.

2. Information of the incoming configurations are supposed to be given from line 100 to line 119. You might want to change these information to match your configurations. The comment there should be quite self-explaining.

3. `make`.

4. Use `mpirun -np $num_node NOARCH.x -qmp-geom $x $y $z $t` to run the job. You should change `$num_node` to match your environment. `$x`, `$y`, `$z` and `$t` are number of nodes to be allocated in each dimensions. The product of these four should be equal to `$num_node`. Examples are

```
mpirun -np 1 NOARCH.x -qmp-geom 1 1 1 1
mpirun -np 256 NOARCH.x -qmp-geom 4 4 4 4
```

5. The results will be placed under `$DBW2-KPI_repo/wflow_tcharge_cps/results/alg_wflow/`.

4.2 Second Stage

1. Go to `$DBW2-KPI_repo`.

2. The input information of this stage is specified in `$DBW2-KPI_repo/conf.sh`. Specify all the information in that file.

3. execute `$DBW2-KPI_repo/do.sh`.

4. The final result will be placed under `$DBW2-KPI_repo/correlation_to_fit/results/`.

4.3 Ensemble Generation

`$DBW2-KPI_repo/ens_gen/binaries` contains the code to generate open/periodic boundary condition pure gauge lattice with DBW2 action. Before compiling you should specify the information of the lattice you want to generate:

1. Lattice size is specified as `int x_sites`, `int y_sites`, `int z_sites` and `int t_sites` in `$DBW2-KPI_repo/ens_gen/vmls/do_arg.vml`.

2. t-direction boundary condition is specsified at line 94 of `$DBW2-KPI_repo/ens_gen/binaries/main.C`.

3. Number of steps in one trajectory and step size are specified as `int steps_per_traj` and `double step_size` in `$DBW2-KPI_repo/ens_gen/vmls/hmc_arg.vml`.

4. β and c_1 of the gauge action are specified as `double beta` and `c_1` in `$DBW2-KPI_repo/ens_gen/vmls/do_arg.vml`. The default value is $c_1 = -1.4088$ for DBW2 action.

5. You are probably safe to use all the other default values.

Now just go to `$DBW2-KPI_repo/ens_gen/binaries`, do `make` and execute the `NOARCH.x` in the same way mentioned in the first stage.

The topological charge density data is automatically generated and placed under `$DBW2-KPI_repo/ens_gen/results/alg_wflow/`. From here you can directly go to the second stage.

References

[1] G. McGlynn and R. D. Mawhinney, Physical Review D - Particles, Fields, Gravitation and Cosmology **90**, 1 (2014), arXiv:1406.4551 .