1 Introduction

Hi, this is the second document about compilation and installation of my code, this time focusing on the Wilson flow suite. So, if you are only using the Wilson flow you do not need to compile the version of FFTW.

My code is written for openmp shared-memory parallelism. And can read NERSC , HiRep, SCIDAC, MILC, `ILDG_SCIDAC` , `ILDG_BQCD` and general LIME configurations.

So, I will now discuss how to compile and run the code. But first a very brief introduction on the Wilson flow.

2 Again a brief introduction to the flow

The Wilson flow is an integration of the flow equation in fictitious time (in practicality, near infinitesimal smearing steps under an RK4 procedure) [**?**][**?**]

$$\partial_t V(t) = Z(V(t))V(t) \qquad V(0) = U. \tag{1}$$

where "U" are the original gauge links, and "Z" is the derivative of the plaquette action at time "t".

What this relates to in practice is (for integration from t to $t+\varepsilon$)

$$\begin{aligned} W_0 &= V(t), \\ W_1 &= \exp\left(\frac{1}{4}Z_0\right)W_0, \\ W_2 &= \exp\left(\frac{8}{9}Z_1 - \frac{17}{36}Z_1\right)W_1, \\ V(t+\varepsilon) &= \exp\left(\frac{3}{4}Z_2 - \frac{8}{9}Z_1 + \frac{17}{36}Z_0\right)W_2. \end{aligned} \tag{2}$$

where $Z_i = \varepsilon Z(W_i)$ and the exponentiation of Z_i is the technique of STOUT smearing, i.e. an exact exponentiation of the hermitian projection of the staples of W_i, simple eh?

2.1 Determining the W_0 value

The idea here is to integrate the respective gauge-action flow until the quantity $t^2\langle E(t)\rangle$ is equal to some value, in fact BMW use the derivative and the well chosen but arbitrary value of 0.3 [**?**].

$$t\partial_t t^2\langle E(t)\rangle|_{t=W_0^2} = 0.3. \tag{3}$$

This is the default stopping value for my code, to change this would require a slight hack in the config.h, I can't imagine why anyone would want to change it.

Also, the energy is defined from the plaquette (or used in practice the **traceless** symmetric clover). As the lattice-ised version of,

$$E(t) = G_{\mu\nu}(t)G_{\mu\nu}(t).$$

On the lattice "G" is the hermitian projection of the clover term or the plaquette.

3 Installation

In the "GLU" folder, you should recognise the usual autoconf stuff, and so compilation should be pretty trouble free. Just the usual configure flags which I will discuss at the bottom.

For the Wilson flow there are two compilation flags available. Which I will describe.

3.1 Improved Clover-term

Ok, so BMW use the simple clover term, which I have to and is the default. But I also have the highly improved $O(a^4)$ clover term from Bilson-Thomson[**?**] (My implementation) to describe the gauge field, this is pretty expensive and I don't often bother using it. As the clover BMW use is fine.

One enables it in the configure script with,

```
--enable-CLOVER_IMPROVE
```

There is a free parameter that can be used in some sense to tune the $O(a^4)$ corrections, this is called K_5. This can be set with the configure argument,

```
--with-CLOVER_K5={}
```

Otherwise the default is $K_5 = 0$.

3.2 Gauge actions

It may be best to use the same staples in the flow as you did for your gauge action, although in the BMW calculation they note that both actions (Wilson and Symanzik) agreed in the continuum limit. This is something I have seen with the Wilson flow and Iwasaki, of course the difference between the two are corrections of $O(a^2)$ and higher so this is to be expected.

I provide the hard-coded option for three common improved gauge actions, they are $IWASAKI, SYMANZIK, D$ the wilson flow (being by far the fastest) is the default. These are enabled with the configure flag,

```
--with-IMPROVED_STAPLE={IWASAKI,SYMANZIK,DBW2}
```

Again, if you want to provide your own plaquette and rectangle weights C_0 and C_1 respectively you can set these by the configure script,

```
--with-IMPROVED_C0={} --with-IMPROVED_C1={}
```

3.3 Compiler flags

If using gcc, I strongly suggest,

```
CFLAGS="-O3 -ffast-math -march=native -fopenmp"
```

3.4 Configure by example

I tested this on my machine and it compiled and ran fine,

```
./configure CFLAGS="-O3 -ffast-math -mfpmath=sse -fopenmp"
--prefix={path to your install dir}
```

and so did the Improved staple and improved field strength tensor.

If configure says the compiler cannot create a simple c program, it doesn't recognise one of the flags.

4 Running my code

Cool, if you are here then that means we are almost done, I will quickly run through the input file.

The input file requires the Mode to be set to "MODE = SMEARING" (unfortunately, the spaces do matter).

Now you want to change "SMEARTYPE", this can be one of the following,

```
WFLOW_STOUT , WFLOW_LOG , ADAPTWFLOW_STOUT , ADAPTWFLOW_LOG
```

the ones with "ADAPT" prefixed are the 2 step adaptive RK4, which is actually faster than the fixed-ε scheme and guarantees accuracy. The suffix "LOG" is for LOG-smearing, which we won't need here but is an interesting case.

The final thing we need is to set "ALPHA1 = 0.01" or some reasonable number (this is what BMW use). Set "SMITERS" to something high like 1000. The algorithm stops if the number of iterations is greater than this or we have reached the W_0 scale or we have reached some integration time specified in `$GLU/src/Smear/adaptive.c`.

4.1 Run commands

My code runs with,

```
./GLU -i input_file -c $CONFIG {-o $outfile}
```

I have emailed you a non-trivial configuration to check this, the output I get is here (run on my desktop) using `WFLOW_STOUT` as the "SMEARTYPE"

The output is the flow time "t" the average plaquette "p" the topological charge "q" the invariant $t^2\langle E(t)\rangle$ "ttGG" and t multiplied by the derivative of this "w".

Note : I only take measurements after t=1, because a flow less than the lattice spacing is not that sensible. Ignore the derivative at t=1. Also note the "FREE-WFLOW" bit, I have two implementations of the Wilson flow, one memory-expensive but quicker and one memory cheap but slower.

References

[1] Luscher, Martin, Trivializing maps, the Wilson flow and the HMC algorithm, Commun.Math.Phys., 293, 899-919, 10.1007/s00220-009-0953-7, 2010, arxiv:/hep-lat/0907.5491, CERN-PH-TH-2009-118.

[2] Luscher, Martin, Properties and uses of the Wilson flow in lattice QCD, JHEP, 1008, 071, 10.1007/JHEP08(2010)071, 2010, arxiv:/hep-lat/1006.4518, CERN-PH-TH-2010-143.

[3] Borsanyi, Szabolcs and Durr, Stephan and Fodor, Zoltan and Hoelbling, Christian and Katz, Sandor D. and others, High-precision scale setting in lattice QCD, JHEP, 1209, 010, 10.1007/JHEP09(2012)010, 2012, arxiv:/hep-lat/1203.4469, ITP-BUDAPEST-657, CPT-P004-2012, WUB-12-02,

[4] , Bilson-Thompson, Sundance O. and Leinweber, Derek B. and Williams, Anthony G., Highly improved lattice field strength tensor, Annals Phys., 304, 1-21, 0.1016/S0003-4916(03)00009-5, 2003, arxiv:/hep-lat/0203008, ADP-01-50-T482.