Lattice Practices 2021

Mattia Dalla Brida

October 2021

1 Definitions

1.1 Gauge action

$$\begin{aligned} S_G[U] &= -\beta \sum_x \mathrm{Re}\{U_{01}(x)\} \\ &= -\beta \sum_x \mathrm{Re}\{U_0(x)U_1(x+\hat{0})U_0(x+\hat{1})^\dagger U_1(x)^\dagger\} \\ &= -\beta \sum_x \cos(\theta_0(x)+\theta_1(x+\hat{0})-\theta_0(x+\hat{1})-\theta_1(x))\,, \end{aligned} \tag{1.1}$$

where we used the relation

$$U_\mu(x) = \exp(i\theta_\mu(x))\,. \tag{1.2}$$

1.2 Fermion action

We consider two degenerate flavours of Wilson-fermions with action

$$S_F[U,\overline{\psi},\psi] = \sum_{f=u,d} \overline{\psi}'_f (D_\mathrm{w}+m_0)\psi'_f\,, \tag{1.3}$$

where

$$D_\mathrm{w} = \sum_{\mu=0}^{1} \tfrac{1}{2}\{\gamma_\mu(\nabla^*_\mu+\nabla_\mu) - a\nabla^*_\mu\nabla_\mu\}\,, \tag{1.4}$$

and

$$a\nabla_\mu\psi(x) = U_\mu(x)^\dagger\psi(x+\hat{\mu})-\psi(x)\,, \tag{1.5}$$

$$a\nabla^*_\mu\psi(x) = \psi(x)-U_\mu(x-\hat{\mu})\psi(x-\hat{\mu})\,. \tag{1.6}$$

In these equations, $\gamma_\mu \equiv \sigma_\mu$, $\mu=0,1$, are the Pauli matrices

$$\sigma_0 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \qquad \sigma_1 = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}\,.$$

With a change of fermion variables, we can recast the action in the form

$$S_F[U,\overline{\psi},\psi] = \sum_{f=u,d} \overline{\psi}_f M(\kappa)\psi_f\,, \tag{1.7}$$

where $M = 1 - \kappa H_{\rm hop}$, $\kappa = 1/(2am_0 + 4)$, and

$$H_{\rm hop} = \sum_{\mu=0}^{1} \delta_{x-\hat{\mu},y}(1+\gamma_\mu)U_\mu(x-\hat{\mu}) + \delta_{x+\hat{\mu},y}(1-\gamma_\mu)U_\mu(x)^\dagger \,. \tag{1.8}$$

2 Hands-on session

Unless differently specified, for the simulations in these exercises you can consider the lattice parameters: $\beta = 4.0$, $\kappa = 0.26$, and $L/a = 12$.

2.1 HMC

1. Generate a thermalized configuration. (Check the available observables for deciding when thermalization seems complete.)

2. Implement a check of the reversibility of the HMC. Starting from some thermalized configuration for $\theta_\mu(x)$, it should measure:

$$\Delta_\theta = ||\theta' - \theta|| = \max_{x,\mu} \left|\theta'_\mu(x) - \theta_\mu(x)\right| , \tag{2.1}$$

$$\Delta_\pi = ||\pi' - \pi|| = \max_{x,\mu} \left|\pi'_\mu(x) - \pi_\mu(x)\right| , \tag{2.2}$$

$$\delta\delta H = |H(\pi',\theta') - H(\pi,\theta)| , \tag{2.3}$$

   where $\tau = n\delta\tau$ is the trajectory length and

$$(\pi',\theta') = F \circ [I_{\rm MD}(\delta\tau)]^n \circ F \circ [I_{\rm MD}(\delta\tau)]^n(\pi,\theta) \qquad F(\pi,\theta) = (-\pi,\theta) \,.$$

   Questions

   - Do you find what you expect?
   - What happens to Δ_θ, Δ_π, and $\delta\delta H$, as τ increases at fixed $\delta\tau$?
   - Play with the solver residuum for the fermionic force computation and see the effect on the reversibility.

3. A proper implementation of the HMC with a sensible choice of algorithmic parameters should satisfy (within errors)

$$\langle e^{-\delta H} \rangle = 1 \,.$$

   Check it.

2.2 MD integration

The code includes an implementation of the OMF 4th order integrator. Based on that:

1. Include in the code the Leap Frog integrator

$$[I_{\rm LPFR}(\delta\tau)]^n = \left(e^{\frac{\delta\tau}{2}\widehat{S}} e^{\delta\tau\widehat{T}} e^{\frac{\delta\tau}{2}\widehat{S}}\right)^n .$$

If you have time, you can also include the Omelyan 2nd order scheme

$$[I_{\mathrm{OMF2}}(\delta\tau)]^n = \left(e^{\lambda\delta\tau\widehat{S}} e^{\frac{\delta\tau}{2}\widehat{T}} e^{(1-2\lambda)\delta\tau\widehat{S}} e^{\frac{\delta\tau}{2}\widehat{T}} e^{\lambda\delta\tau\widehat{S}}\right)^n .$$

For a more efficient implementation one should combine the force calculation at intermediate steps, e.g.

$$[I_{\mathrm{LPFR}}(\delta\tau)]^n = e^{\frac{\delta\tau}{2}\widehat{S}} e^{\delta\tau\widehat{T}} \underbrace{e^{\frac{\delta\tau}{2}\widehat{S}} e^{\frac{\delta\tau}{2}\widehat{S}}}_{e^{\delta\tau\widehat{S}}} e^{\delta\tau\widehat{T}} e^{\delta\tau\widehat{S}} \cdots e^{\delta\tau\widehat{S}} e^{\delta\tau\widehat{T}} e^{\frac{\delta\tau}{2}\widehat{S}}$$

How many force computations do we have over a trajectory for the different integrators (LPFR, OMF2, OMF4)?

2. Study the dependence of δH for the different integrators as a function of $\delta\tau$ at fixed τ. You can do a single trajectory starting from the very same gauge-configuration and using the same seed, while changing the number of steps. Otherwise, you can measure $\langle|\delta H|\rangle$ over many configurations while changing the number of steps. In this case, the initial configuration and seed do not matter (as long as you are thermalized). You can consider the pure gauge theory for a quicker test. Also note that $\delta H = \mathrm{O}(\delta\tau^n)$ only asymptotically for $\delta\tau \to 0$. You must, thus, find the range of $\delta\tau$ where the asymptotic behavior kicks in.

3. What integrator do you find to be the most cost effective at these physical parameters? How would you assess that?

2.3 Topology freezing

To experience the issue with topology freezing we can study the continuum limit of the topological charge in pure U(1) gauge theory. In order to detect some non-trivial topology, the physical volume of our lattice should not be too small. For this reason we take $R = V/\beta = 80$; note that the (inverse) bare coupling β is dimensionful in the $2d$ Schwinger model. The continuum limit is approached by taking $\beta \to \infty$ at fixed R. You can consider for instance simulations at the bare parameters given in Table 1. Plot the Monte Carlo history of the geometrical definition of the topological charge:

$$Q \equiv \frac{1}{2\pi} \sum_x \mathrm{Im}\ln(U_{01}(x)) ,$$

which should assume integer values. What do you observe?

2.4 Forces

1. To understand the benefits of Hasenbusch preconditioning compare the performances of a run with a single pseudofermion and one with two, taking for this case $\mu_1 \approx 0.1$. What do you observe?

2. Have a look at the norm of the fermionic forces for the two cases. What does Hasenbusch preconditioning achieve?

3. Have also a look at the size and cost of the different forces. How do the gauge and fermionic forces compare?

β	L/a
5	20
6.05	22
7.2	24
8.45	26
9.8	28
11.25	30

Table 1: Line of constant physics for the pure $U(1)$ gauge theory corresponding to $V/\beta = 80$.

3 Code

The main program[1] can be found in `main/schwinger.c` and it is compiled by typing `make` in `main/`. The way to execute the program is:

```
schwinger input cnfg > output
```

Examples of input files for QED_2 and the pure-gauge theory are given in `main/infile_qed` and `main/infile_ym`, respectively. Check also `main/infile_readme` for a definition of the different input parameters. The starting configuration `cnfg` is an optional argument. If it is not specified, the code starts from the "cold" configuration: $\theta_\mu(x) = 0$.

The size of the lattice can be changed in `include/lattice.h`. Note that the code does not support dimensions $D \neq 2$. In this file are also defined the two global fields which can be accessed from every module in the package. These are the global gauge field `gauge[ix][mu]` and hopping matrix `hop[ix][mu]`; for a description of the later consult `modules/geometry/hopping.c`. The points of the lattice are indexed in lexicographic order: `ix` $= \sum_{\mu=0}^{D-1} x_\mu L^\mu$ and `hop[ix][mu]` contains the information about the nearest neighbors of each point. The main routines to complete are found in `modules/update/hmc.c`, which is the core of the program. Specifically, these are: `leapfrog()`, `omelyan2()`, `reversibility()`, and `flip_mom()`.

For the development of the check on reversibility you may consider using a separate main file which you can find in `devel/check/check1.c`.

By adding the option `-DMDINT_DBG` to the `CFLAGS` in `main/Makefile` you activate the computation and printout of the module of the HMC forces and their timings (see `modules/update/hmc.c` for more details).

[1] The code is a customized version of the `schwinger` package written by Stefan Schaefer for the Lattice Practices 2018 Edition. We thank the author of the code for shearing this with us, as well as for his useful comments on the exercises.