Wilson loop calculation

Jornadas do CeFEMA

Alireza Sharifian

Instituto Superior Técnico

16.Sep.2019

what can we extract:

- lattice

Figure: 3D lattice

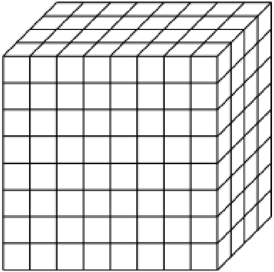
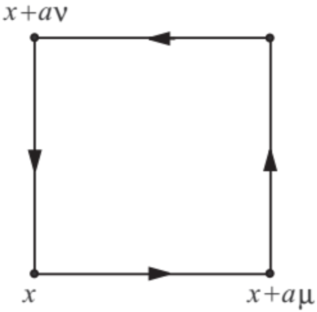


Figure: Plaquette p in direction μ and ν



- Expectation value of any observable :

$$\langle A \rangle = \frac{1}{Z} \int \prod_b dU(b)\ A\ e^{-S_W} \qquad \int dU = 1.$$

Wilson Loop

- Wilson loop is defined as:

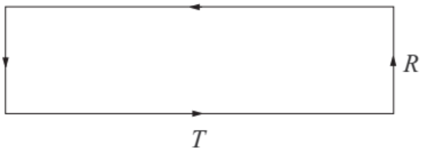


Figure: wilson loop

- for $T \to \infty$ *we have*
  $\langle tr(U(\mathcal{C})\rangle = \exp(TV(R))$
  $V(R) =$ *static quark-antiquark potential*

Plaquette-plaquette correlation

- correlation of two spacial plaquette

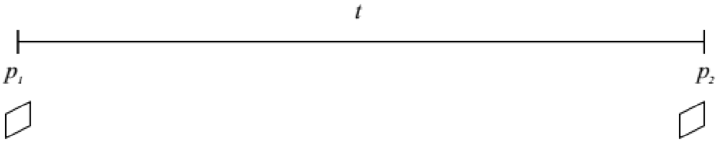


Figure: 2 spatial plaquette $p_1$and $p_2$separated time t

$\langle Tr(U(p_1))Tr(U(p_2))\rangle = \exp(-mt)$
$m =$ lowest particle mass in the theory(gluebal)[3]

Numerical Calculation

- extraction of effective mass diagram
  $\langle w(r,t)\rangle$ = average of wilson loops with size $r \times t$
  $r = n\,a, t = n_t a$
  $\langle w(r,t)\rangle = \mathcal{C}\exp(n_t a V(na))$

$$aV(na) = \log\left(\frac{w(na, n_t a)}{w(na, (n_t+1)a)}\right)$$

  $V(r) = A + \frac{B}{r} + \sigma r \qquad F(r) = \frac{d}{dr}V(r)$
- From experimentat data and Schrodinger equation
  $F(r_0)r_0^2 = -B + \sigma r_0^2 = 1.65$
  $\frac{r_0}{a} = \sqrt{\frac{1.65+B}{\sigma a^2}}$

$$aV(na) = Aa + \frac{B}{n} + \sigma a^2 n$$

- jackknife

$$\sigma_{\hat{\theta}}^2 = \frac{N-1}{N}\sum_{n=1}^{N}(\theta_n - \hat{\theta})^2$$

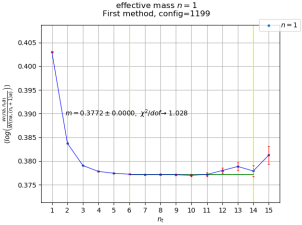
Numerical results

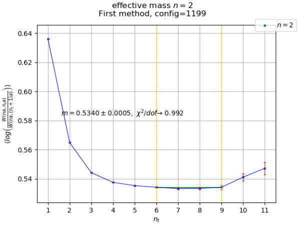
- Value of wilson loops

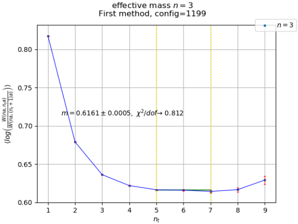
```
w(6, 9)=-0.00007645689363
w(6, 10)=0.00012333223216
w(6, 11)=-0.00022383341051
w(6, 12)=-0.00013875277183
w(6, 13)=0.00023426581602
w(6, 14)=-0.00011709960653
w(6, 15)=0.00033422974705
w(6, 16)=0.00010538573652
w(7, 1)=0.06124080095628
w(7, 2)=0.01444496191508
w(7, 3)=0.00510581476991
w(7, 4)=0.00221613904016
w(7, 5)=0.00132391459250
w(7, 6)=0.00073946553723
w(7, 7)=0.00000450756059
w(7, 8)=-0.00010455638400
w(7, 9)=-0.00004821119003
w(7, 10)=0.00010488054397
w(7, 11)=0.00006249572781
w(7, 12)=-0.00010205010220
w(7, 13)=0.00017483009148
w(7, 14)=0.00005278947736
w(7, 15)=0.00000758609606
w(7, 16)=-0.00009137646582
```

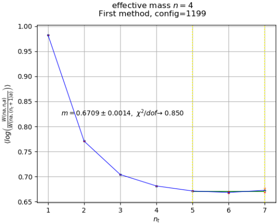
Results

- effective mass plots



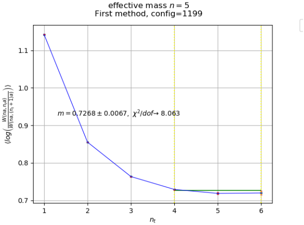


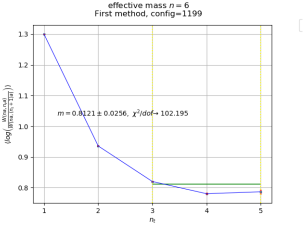


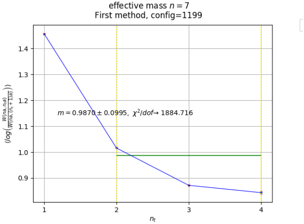


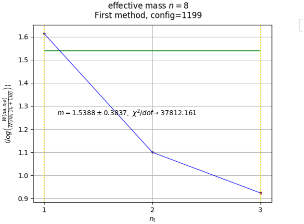
Effective mass plot

- $5 \leq r \leq 8$









Quark Potential

- properties of the lattice

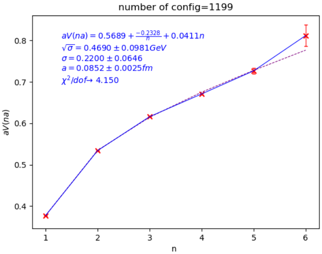


Figure: Potential

Comparision of speed and code

- speed(s), 4 Core

	lattice rearrangment	wilson line	wilson loop
CPU	0.25	6.5	230
Parallel-CPU	0.085	3	77
GPU	-	0.03	1.5

- cpu properties:

```
vendor_id       : GenuineIntel
cpu family      : 6
model           : 30
model name      : Intel(R) Core(TM) i7 CPU        860  @ 2.80GHz
```

```
Device 1: "GeForce GTX TITAN Black"
  CUDA Driver Version / Runtime Version          10.1 / 10.1
  CUDA Capability Major/Minor version number:    3.5
  Total amount of global memory:                 6083 MBytes (6378749952 bytes)
  (15) Multiprocessors, (192) CUDA Cores/MP:     2880 CUDA Cores
  GPU Max Clock rate:                            980 MHz (0.98 GHz)
  Memory Clock rate:                             3500 Mhz
  Memory Bus Width:                              384-bit
```

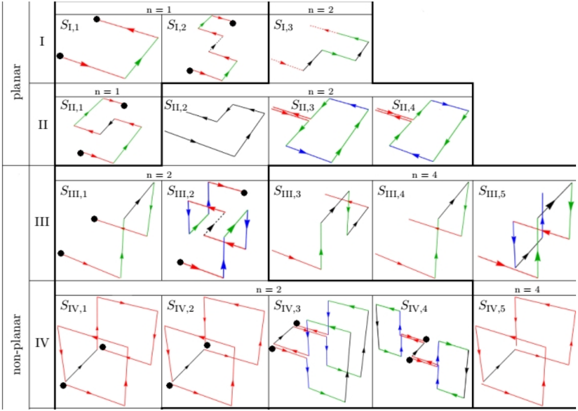
how to change the code for omp

```
auto t1=get_time::now();
            for (int t = 0; t < s[3]; t++)
                for (int k = 0; k < s[2]; k++)
                    for (int j = 0; j < s[1]; j++)
                        for (int i = 0; i < s[0]; i+
                            for (int dir = 0; dir <
                                for (int p = 0; p <
                                    for (int q = 0;
                                        OneDArray[ne
                                    }
#pragma omp parallel
        {
            double t1, t2;
            t1 = omp_get_wtime();
#pragma omp for
            for (int t = 0; t < s[3]; t++)
                for (int k = 0; k < s[2]; k++)
                    for (int j = 0; j < s[1]; j++)
                        for (int i = 0; i < s[0]; i+
                            for (int dir = 0; dir <
                                for (int p = 0; p <
                                    for (int q = 0;
                                        OneDArray[ne
                                    }
```

Figure: 0.25 to 0.085 ms

Next Phase

- calculation of more complecated wilson loops[1]



Stefano Capitani, Owe Philipsen, Christian Reisinger, Carolin Riehl, and Marc Wagner.
Precision computation of hybrid static potentials in SU(3) lattice gauge theory.
Phys. Rev., D99(3):034502, 2019.

Christof Gattringer and Christian B. Lang.
Quantum chromodynamics on the lattice.
Lect. Notes Phys., 788:1–343, 2010.

G. Munster and M. Walzl.
Lattice gauge theory: A Short primer.
In *Phenomenology of gauge interactions. Proceedings, Summer School, Zuoz, Switzerland, August 13-19, 2000*, pages 127–160, 2000.

Acknowledgements

- I would like to thanks Dr. Nuno Cardoso for his kind helps to me for learning CUDA.
- I acknowledge the financial support by FCT.