Lattice field theories
Exercise sheet 6 – Wilson loop and string tension

Lectures: Jan Pawlowski j.pawlowski@thphys.uni-heidelberg.de
Felipe Attanasio pyfelipe@thphys.uni-heidelberg.de

Tutorials: Felipe Attanasio pyfelipe@thphys.uni-heidelberg.de

Institut für Theoretische Physik, Uni Heidelberg

Please see the supplemental material at the end of this exercise sheet for further explanations and hints.

Exercise 11: Connecting strong and weak coupling

Consider a 2-dimensional SU(2) Yang-Mills theory on the lattice, whose partition function reads

$$Z = \int \mathcal{D}U \, \exp\left\{ -\beta \sum_n \left[1 - \frac{1}{2} \mathrm{ReTr} U_{01}(n) \right] \right\} , \tag{1}$$

where U_{01} is the plaquette at site n and directions 0 and 1. The Wilson loop defined in a contour $\mathcal{C}$ of size $T \times R$ lattice site n is given by

$$W(\mathcal{C}) \equiv W(T, R) = \frac{1}{2} \left\langle \mathrm{Tr} \left[\prod_{\mathcal{C}} U \right] \right\rangle . \tag{2}$$

Use the Metropolis algorithm to simulate this theory on a 10×10 lattice with $0.1 \leq \beta \leq 15.0$, and compute the Wilson loop for square contours of different sizes.

You should observe that the 1×1 Wilson loop has the following limiting behaviours:

- at strong coupling ($\beta \ll 1$):

$$W(1,1) \underset{\beta \to 0}{\sim} \frac{\beta}{4} ; \tag{3}$$

- at weak coupling ($\beta \gg 1$):

$$W(1,1) \underset{\beta \to \infty}{\sim} 1 - \frac{3}{2\beta} . \tag{4}$$

You should also note that larger Wilson loops are suppressed at small β, and that getting good sinals for them can be hard. You may want to run longer simulations than in the ϕ^4 case.

Hint: In order to improve statistics, consider computing $W(1,1)$ on all lattice points and then taking the volume average. Do not forget about the periodic boundary conditions.

Hint: For large β configurations where plaquette is close to the identity will become more and more important. Consider reducing the variance of your proposals in these situations, e.g., using a narrower Gaussian to generate random numbers, to achieve a higher acceptance rate.

Exercise 12: Computing the string tension

From strong coupling arguments, we expect the above Wilson loop to have the form

$$W(T,R) = \exp\left[-\hat{\sigma}RT - \hat{\alpha}(R+T) + \hat{\gamma}\right] , \tag{5}$$

where $\hat{\sigma}$ is the *string tension* in lattice units, and $\hat{\alpha}$ accounts for subleading effects proportional to the perimeter of the contour. Those subleading effects can be eliminated by studying the *Creutz ratio*,

$$\hat{\chi}(T,R) = -\ln\left(\frac{W(T,R)\,W(T-1,R-1)}{W(T-1,R)\,W(T,R-1)}\right) . \tag{6}$$

If eq. (5) correctly describes the behaviour of the Wilson loop, then $\hat{\chi}$ will coincide with the string tension.

In a way similar to the previous exercise, compute Wilson loops of square and rectangular shapes and construct Creutz ratios for various values of β. Like before, ratios computed at larger values of T or R will be noisier. Since we expect $\hat{\chi}$ to be independent of both T and R, a constant fit (including errors, obviously!) will give the final result.

At Strong coupling, your string tension should behave as

$$\hat{\sigma} \underset{\beta\to 0}{\sim} -\ln\frac{\beta}{4} . \tag{7}$$

Hint: Due to larger Wilson loops being suppressed at small β their values may come out negative, but statistically compatible with zero and probably large relative errors. Since the string tension is a real number, taking the logarithm of negative numbers makes no sense. In such situations, you can either disregard the Creutz ratios where W is negative, or run your simulation for longer.

Hint: In order to improve statistics, consider computing $W(T,R)$ on all lattice points and then taking the volume average. Do not forget about the periodic boundary conditions.

Hint: When computing $\hat{\chi}(1,1)$ use only $W(1,1)$ in the calculation, i.e., take the other loops to be equal to 1.

Supplemental material

In this exercise we are dealing with a field whose elements belong to the group SU(2), and the updating procedure should be such that update proposals also belong to the group.

Since every element of SU(2) group can be written as the exponential of a member of the $\mathfrak{su}(2)$ Lie algebra, updating a particular link U can be written as

$$U \to R\,U\,, \quad R = \exp\left[i\sum_{a=1}^{3}\alpha^a\sigma^a\right]\,, \tag{8}$$

with σ^a being the Pauli matrices and α^a real numbers. Note that the update may equally well be written as $U \to UR$. Because of the commutation properties of the Pauli matrices the computation of R can be simplified as

$$R = \cos(|\vec{\alpha}|)\mathbb{1} + i\frac{\alpha^a}{|\vec{\alpha}|}\sigma^a\sin(|\vec{\alpha}|)\,, \tag{9}$$

with $\mathbb{1}$ being the 2×2 identity matrix and $|\vec{\alpha}| = \sqrt{(\alpha^1)^2 + (\alpha^2)^2 + (\alpha^3)^2}$. Thus it requires only simple arithmetic and trigonometric functions.

```
matrix U[Nt][Nx];
for n ← 0 to N_steps do
    for 0 ≤ t < N_t and 0 ≤ x < N_x do              // sweep over the lattice
        oldAction ← S(U) ;                                // save old action
        oldValue ←U[t][x] ;                                  // save old U
        R ← randomSU2() ;                              // random SU(2) matrix
        U[t][x] ← R * U[t][x] ;     // random change to U within SU(2)
        ΔS = S(U) - oldAction;
        if ΔS > 0 then
            r ← UniformReal(0,1);
            if r > exp(−ΔS) then
                U[t][x] ← oldValue ;                      // reject proposal
            end
        end
    end
    Obs ← Obs + O(U) ;               // update computation of observable
end
Obs ← Obs / N_steps;
```

As always, you should skip a number of steps at the beginning, so the system can thermalise. And remember that consecutive configurations in the Markov chain are likely to be correlated, so it is also a good idea to skip some between consecutive measurements.

Below, you can find a plot of the string tension as a function of β to use for cross-checking.

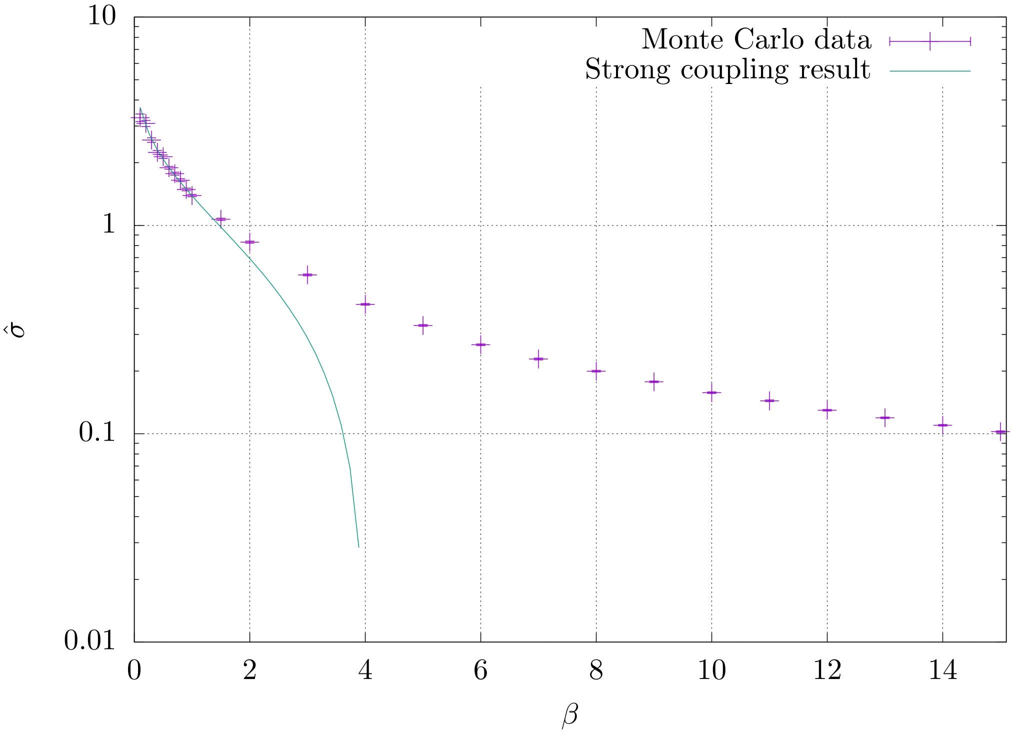

Figure 1: String tension, in lattice units, as a function of the (inverse) coupling. Computed on a 10×10 lattice.