Report for the Course Modelling in Computational Science, HT23

Project 1: The Potts model

Theo Koppenhöfer

Lund

September 14, 2023

Introduction

The following report is part of the first project of the Modelling in Computational Science course at Lund University, HT2023. In the first project we implemented a Monte-Carlo simulation of the q-state Potts model. The report and the Python implementation can be found online under [1]

The Potts model

A state of the q-state Potts model of a $L \times L$ flat grid is given by a mapping

$$s\colon \{1,\dots,L\} \times \{1,\dots,L\} \to \{1,\dots,L\}\,.$$

One assigns this state an energy via

$$E = -J \sum_{\substack{i,j \text{ neighbouring} \\ \text{grid points}}} \delta(s_i, s_j)$$

where δ denotes the Kronecker-delta and $J = 1$ is the coupling strength. As given in the problem setting we assume periodic boundary conditions on the grid.

The implementation

Determining when the Energy has plateaued

The energy of the system takes some time t_0 to reach an equilibrium state. To determine t_0 we calculate in every step i moving averages ma_1 and ma_2 over n energies. The construction is shown in figure 1. If we start in the hot state then the energy will tend to shrink until we reach an equilibrium. Hence we can use the condition

$$\mathrm{ma}_2 \leq \mathrm{ma}_1 \tag{1}$$

to determine t_0. If on the other hand we start in cold state the energy will in general increase and we have to reverse the inequality in equation (1). After reaching an equilibrium the simulation runs for a further `M_sampling=5000` steps. It is over these samples we take the mean and the variance.

Improving performance

While simulating we run into performance issues due to the fact that Python is in general rather slow even with heavy use of Numpy. We resolved this issue with the help of Numba which precompiled functions and thus increased performance substantially. The downside is that the code becomes quite unestetic because Numba only supports some Python and Numpy features. We noted during the simulation of the

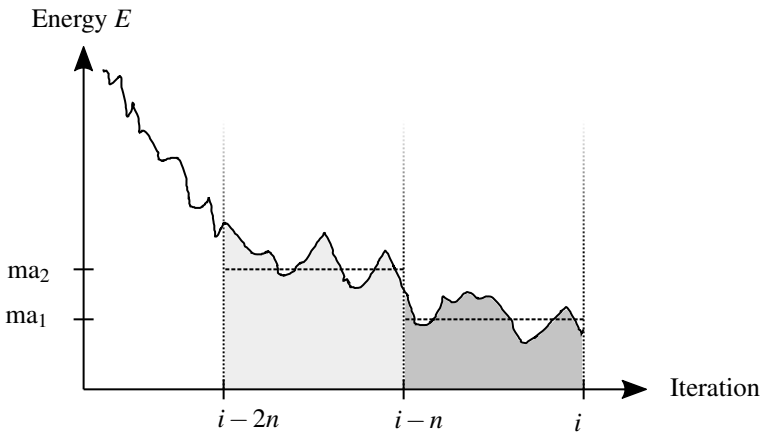
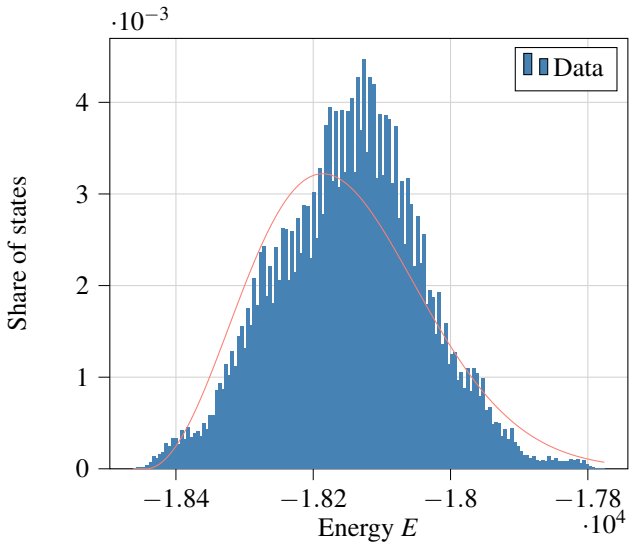
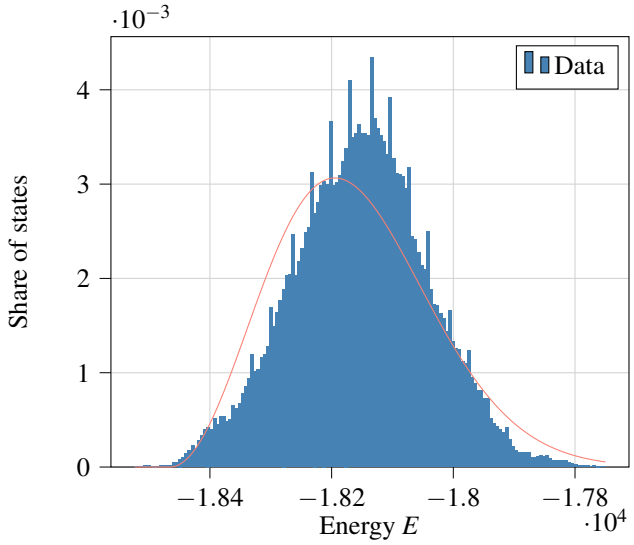
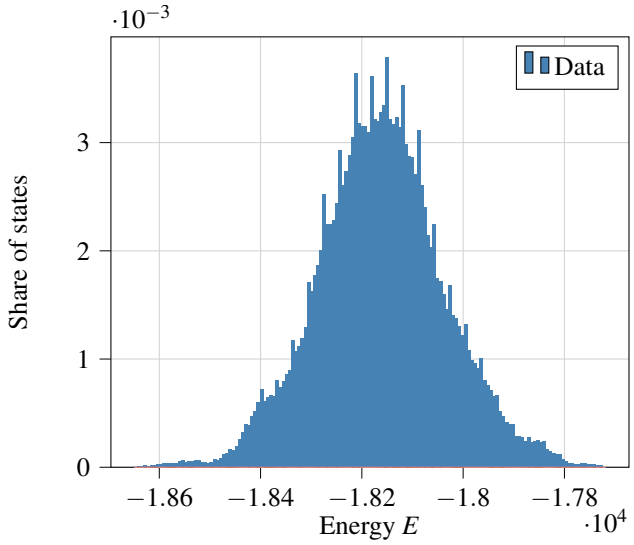
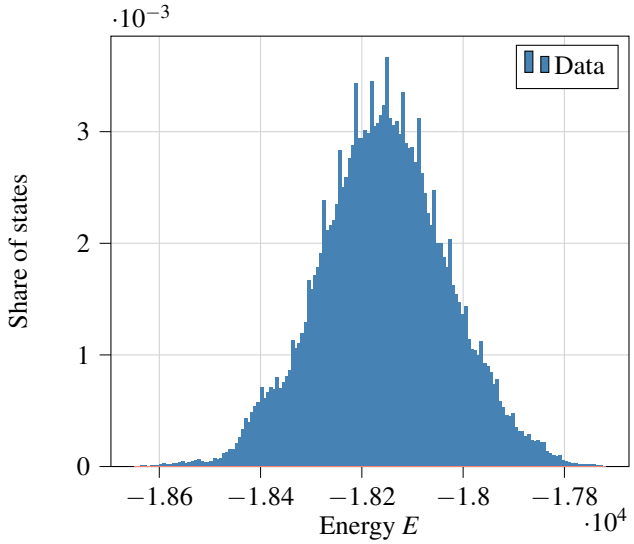


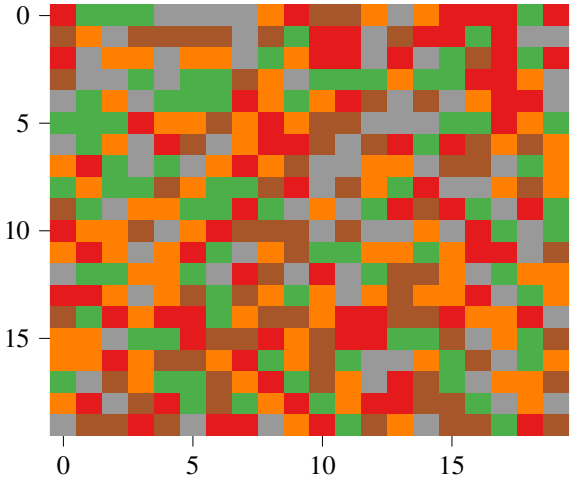
Figure 1: Explanation of the moving averages.




$\cdot 10^{-3}$
Data
Share of states
4
3
2
1
0
−1.84
−1.82
−1.8
−1.78
Energy E
$\cdot 10^4$


$\cdot 10^{-3}$
Data
Share of states
3
2
1
0
−1.86
−1.84
−1.82
−1.8
−1.78
Energy E
$\cdot 10^4$


$\cdot 10^{-3}$
Data
3
2
1
0
Share of states
−1.86
−1.84
−1.82
−1.8
−1.78
Energy E
$\cdot 10^4$


0
5
10
15
0
5
10
15

Bibliography

[1] computational-science-HT23, *Github repository to the project.* Online, 2023. [Online]. Available: `https://github.com/TheoKoppenhoefer/computational-science-HT23`.