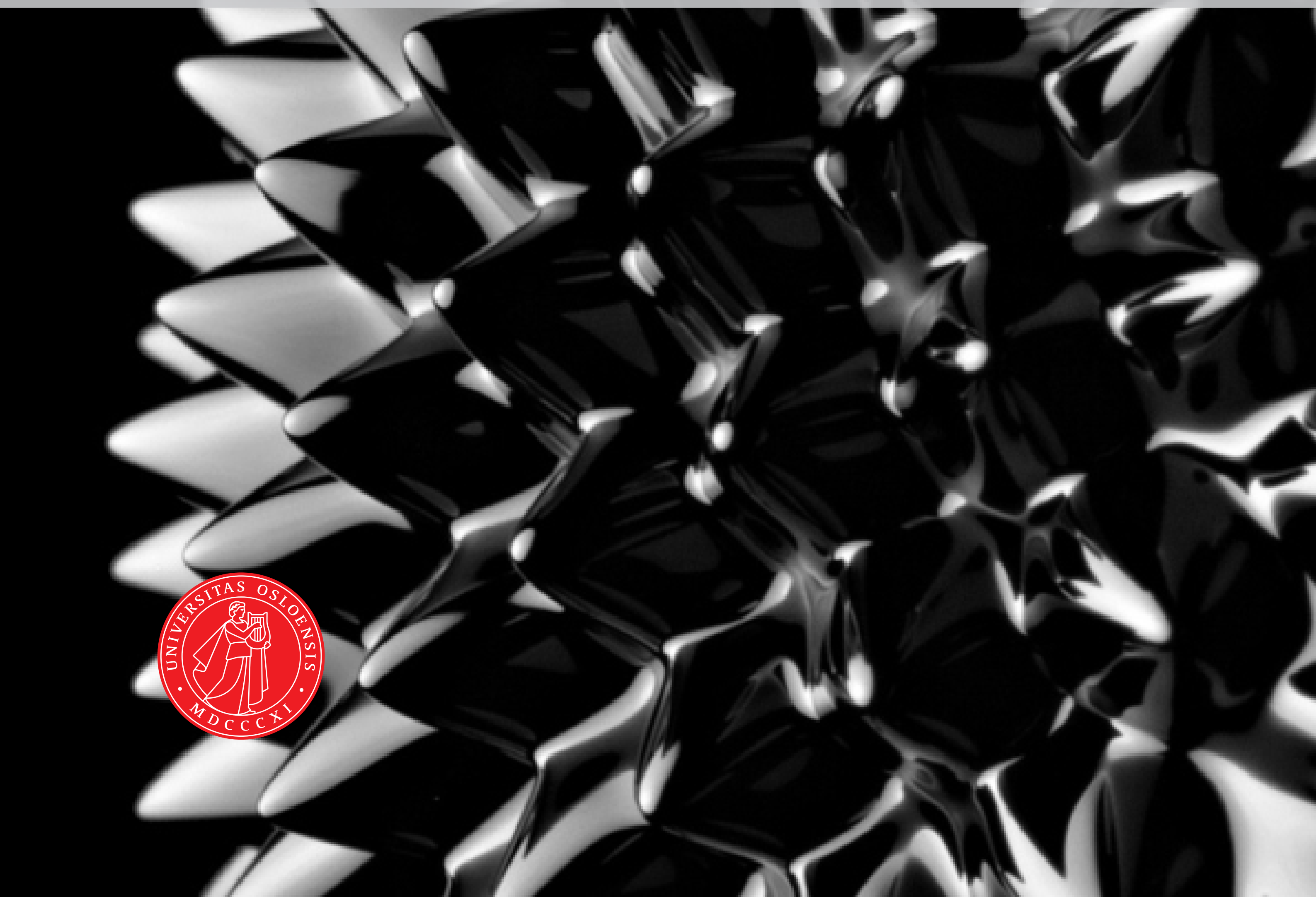
UiO : Department of Physics
University of Oslo

Studies of phase transitions in magnetic systems

Erik Skaar



Contents

Abstract

1 Introduction

These laws are not enough to solve the motion of the planets. From the laws one can derive differential equations for the motion, which are not trivial or even possible to solve analytically. This is where computational methods are useful. With the tools developed in computtational physics we can make a prediction to the motion of the planets in our solar system.[1] And because of our assignment we kind of have to do this to pass the course.[2]

[1] Semester page for FYS3150 - Autumn 2017.

2 Theory

2.1 the Ising model

The Ising model describes a coupled system. Where only the nearest neighbor affect each other. In this report the Ising model will be applied to a two dimensional magnetic system. This will be a grid of spins, where each spin s_i can either have 1 or 0 as value. The total energy is expressed as:

$$E = -\sum_{<i,j>} J_{i,j} s_i s_j$$

Where the symbol $<$ kl $>$ indicates that we sum over nearest neighbors only. If we assume that each coupling has the same magnitude J, then the energy is expressed as:

$$E = -J \sum_{<i,j>} s_i s_j \tag{1}$$

2.1.1 Periodic boundary conditions

When working with a finite matrix we run into a problem with the boundaries. They are missing neighbours. We solve this by introducing periodic boundary conditions. This means that the right neighbour for S_n is assumed to take the value of S_1.

2.2 Statistical physics

2.2.1 the partition function

Boltzmann distrubtion is used as the probability distrubtion. Boltzmann distribution states the probability for E_i is proportional to $e^{-\beta E_i}$, where β is $\frac{1}{k_B T}$. k is the boltzmann constant. For this to be a probability distribution, it needs to be normalized. To normalize the distribution divide the sum of probabilities by a constant Z:

$$1 = \frac{\sum_i e^{-\beta E_i}}{Z}$$
$$Z = \sum_i e^{-\beta E_i}$$

Z is called the partition function.

2.2.2 Calculation of values

The partition function is very useful. In combination with the boltzmann distrubution we get a expression for the probability.

$$P(E_i) = \frac{e^{-\beta E_i}}{Z}$$

For finding a mean value, one can simply make a sum over $P(E_i)$ multiplied by the value of interest. For instance the mean energy is given by:

$$\langle E \rangle = \sum_i E_i P(E_i)$$

Expressions for important expectation values can be derived such for the energy E, magnetic moment |M|, specific heat capacity C_v and the sesceptibility χ. The expressions used in this report are listed below[1]:[2]

$$\langle E \rangle = \sum_i E_i P(E_i) \tag{2}$$

$$\langle |M| \rangle = \sum_i M_i P(E_i) \tag{3}$$

$$\langle C_V \rangle = \frac{1}{kT^2}\left(\langle E^2 \rangle - \langle E \rangle^2\right) \tag{4}$$

$$\langle \chi \rangle = \frac{1}{kT}\left(\langle M^2 \rangle - \langle |M| \rangle^2\right) \tag{5}$$

2.3 Phase transition

2.4 Randomness

[2] lecture note page 427

3 Method

4 Result & Discussion

4.1 Analytic 2x2

4.1.1 Microstates 2x2

Table 1: This shows the different microstates that is possible for a 2x2 spinmatrix. It also states the energy and magnetic moment for each microstate.

State	Energy	Magnetic moment	State	Energy	Magnetic moment
$\begin{smallmatrix}\uparrow & \uparrow \\ \uparrow & \uparrow\end{smallmatrix}$	-8J	4	$\begin{smallmatrix}\downarrow & \downarrow \\ \downarrow & \downarrow\end{smallmatrix}$	-8J	-4
$\begin{smallmatrix}\downarrow & \uparrow \\ \uparrow & \uparrow\end{smallmatrix}$	0J	2	$\begin{smallmatrix}\uparrow & \downarrow \\ \downarrow & \downarrow\end{smallmatrix}$	0J	-2
$\begin{smallmatrix}\uparrow & \downarrow \\ \uparrow & \uparrow\end{smallmatrix}$	0J	2	$\begin{smallmatrix}\downarrow & \uparrow \\ \downarrow & \downarrow\end{smallmatrix}$	0J	-2
$\begin{smallmatrix}\uparrow & \uparrow \\ \downarrow & \uparrow\end{smallmatrix}$	0J	2	$\begin{smallmatrix}\downarrow & \downarrow \\ \uparrow & \downarrow\end{smallmatrix}$	0J	-2
$\begin{smallmatrix}\uparrow & \uparrow \\ \uparrow & \downarrow\end{smallmatrix}$	0J	2	$\begin{smallmatrix}\downarrow & \downarrow \\ \downarrow & \uparrow\end{smallmatrix}$	0J	-2
$\begin{smallmatrix}\downarrow & \downarrow \\ \uparrow & \uparrow\end{smallmatrix}$	0J	0	$\begin{smallmatrix}\uparrow & \uparrow \\ \downarrow & \downarrow\end{smallmatrix}$	0J	0
$\begin{smallmatrix}\downarrow & \uparrow \\ \downarrow & \uparrow\end{smallmatrix}$	0J	0	$\begin{smallmatrix}\uparrow & \downarrow \\ \uparrow & \downarrow\end{smallmatrix}$	0J	0
$\begin{smallmatrix}\uparrow & \downarrow \\ \downarrow & \uparrow\end{smallmatrix}$	8J	0	$\begin{smallmatrix}\downarrow & \uparrow \\ \uparrow & \downarrow\end{smallmatrix}$	8J	0

Table 2: The table shows a summary from table 4.1.1.

Number of $\uparrow$	Multiplicity	Energy	Magnetic moment
4	1	-8J	4
3	4	0J	2
2	2	8J	0
2	4	0J	0
1	4	0J	-2
0	1	-8J	-4

4.1.2 Quantities

We will use the equations from section 2.2.2.

For energy the eq. 2 will result in:

$$
\begin{aligned}
&Z = \sum_i = e^{-\beta E_i} \\
&\mathrm{T} = \mathrm{kT/J} = 1 \\
&Z = \sum_i = e^{-\beta E_i} = 2e^8 + 2e^{-8} + 12
\end{aligned}
$$

For energy the eq. 2 will give the result:

$$
\begin{aligned}
&\langle E\rangle = \sum_i E_i P(E_i) \\
&\mathrm{T} = \mathrm{kT/J} = 1 \\
&\langle E\rangle = \frac{1}{Z}\sum_i E_i e^{-E_i} \\
&\langle E\rangle = \frac{1}{Z}\left(16e^8 - 16e^{-8}\right) = 7.9839 \\
&\langle E\rangle/N = \frac{\langle E\rangle}{4} = 1.9959
\end{aligned}
$$

For energy the eq. 3 will give the result:

$$
\begin{aligned}
&\langle |M|\rangle = \sum_i M_i P(E_i) \\
&\mathrm{T} = \mathrm{kT/J} = 1 \\
&\langle |M|\rangle = \frac{1}{Z}\sum_i M_i e^{-E_i} \\
&\langle |M|\rangle = \frac{1}{Z}\left(4\cdot 1e^8 + 2\cdot 4e^0 + 0\cdot 2e^{-8} + 0\cdot 4e^0 + 2\cdot 4e^0 + 4\cdot 1e^8\right) \\
&\langle |M|\rangle = \frac{1}{Z}\left(16 + 8e^8\right) = 3.9946 \\
&\langle |M|\rangle/N = \frac{\langle M\rangle}{4} = 0.9986
\end{aligned}
$$

For C_V we need to calculate $\langle E^2\rangle$:

$$
\begin{aligned}
\langle E^2 \rangle &= \sum_i E_i P(E_i) \\
\mathrm{T} &= \mathrm{kT/J} = 1 \\
\langle E^2 \rangle &= \frac{1}{Z} \sum_i E_i^2 e^{-E_i} \\
\langle E^2 \rangle &= \frac{1}{Z} \left(128e^8 + 128e^{-8}\right) \\
C_V &= \langle E^2 \rangle - \langle E \rangle^2 = 0.12832 \\
C_V/N &= 0.03208
\end{aligned}
$$

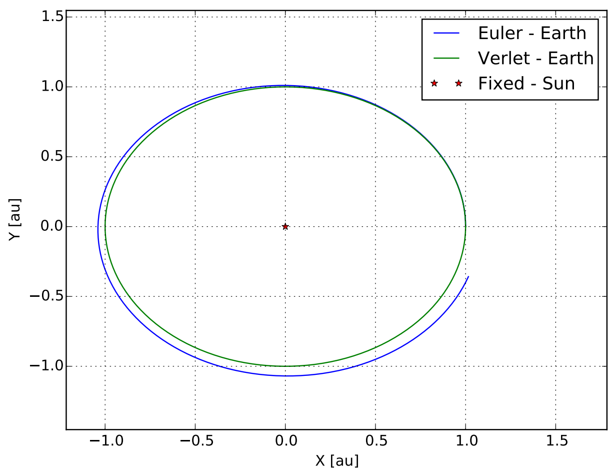
For χ we need to calculate $\langle M^2 \rangle$:

$$
\begin{aligned}
\langle M^2 \rangle &= \sum_i M_i^2 P(E_i) \\
\mathrm{T} &= \mathrm{kT/J} = 1 \\
\langle M^2 \rangle &= \frac{1}{Z} \sum_i M_i^2 e^{-E_i} \\
\langle M^2 \rangle &= \frac{1}{Z} \left(16 \cdot 1e^8 + 4 \cdot 4e^0 + 0 \cdot 2e^{-8} + 0 \cdot 4e^0 + 4 \cdot 4e^0 + 16 \cdot 1e^8\right) \\
\langle M^2 \rangle &= \frac{1}{Z} \left(32 + 32e^8\right) = 15.9732 \\
\langle M \rangle & \text{ is 0 which makes } \langle \chi \rangle = \langle M^2 \rangle \\
\langle \chi \rangle &= 15.9732 \\
\langle \chi \rangle/N &= 3.9933
\end{aligned}
$$

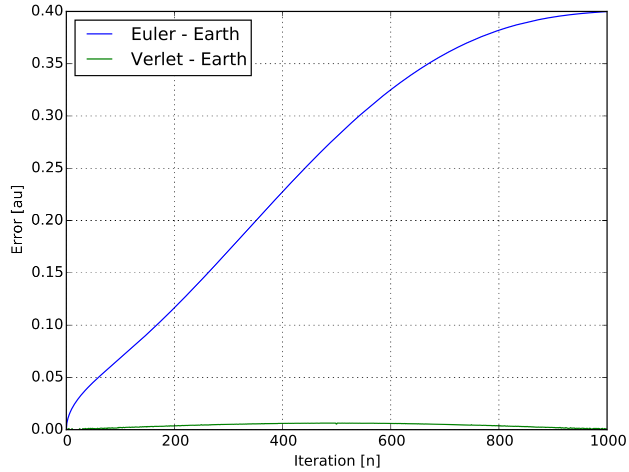
Below you can see a summary for the quantities:

$$
\begin{aligned}
\langle E \rangle/N &= 1.9959 & \langle |M| \rangle/N &= 0.9986 \\
C_V/N &= 0.03208 & \langle \chi \rangle/N &= 0.004010
\end{aligned}
$$

4.2 example



(a)



(b)

Figure 1: a) shows the orbit of earth around the sun. The intial velocity is set to 2π in y direction and the start position to 1 au in x direction. b) shows how the error develops. The intial values should give a perfect circular motion. So the error is calculated by $r_i - r_0$. It is apparent that the Verlet-Velocity method is a better approximation. This simulation was with 1000 points with the end time of 1 year. Both simulations was produced by plot_earth_sun.py

5 Conclusion

6 References

References

[1] Morten Hjorth-Jensen. *Computational Physics*. Lecture notes. 2015. URL: https://github.com/CompPhysics/ComputationalPhysics/blob/master/doc/Lectures/lectures2015.pdf.

[2] Morten Hjorth-Jensen. *Computational Physics*. Project-4. 2017. URL: https://github.com/CompPhysics/ComputationalPhysics/blob/master/doc/Projects/2017/Project4/pdf/Project4.pdf.

7 Appendix

```
//FLOPs FOR POSITION :: EULER
// 2 FLOPs * 3 directions
x = x + t_step*Vx
//TOTAL FLOPs = 6 FLOPs
```