Diplomarbeit in Physik

Atomistic Spin-Dynamics
in Confined Magnetic Nano-Structures

von

David Bauer

vorgelegt der

Fakultät für Mathematik, Informatik und Naturwissenschaften
der Rheinisch-Westfälischen Technischen Hochschule Aachen

im
Oktober 2008

angefertigt am

Institut für Festkörperforschung (IFF)
Forschungszentrum Jülich

Contents

1 Introduction **5**

2 Landau-Lifschitz Dynamics **9**
2.1 Introduction . . . 9
2.2 Hamiltonian . . . 10
2.2.1 Heisenberg Hamiltonian . . . 10
2.2.2 Zeeman Term . . . 12
2.2.3 Magnetocrystalline Anisotropy . . . 12
2.2.4 Dipole-Dipole-Interaction . . . 14
2.3 Classical Approach . . . 15
2.3.1 Classical Equation of Motion . . . 16
2.3.2 Simulation of Damping . . . 18
2.4 Temperature Bath . . . 20
2.4.1 Langevin-Like Approach . . . 21
2.4.2 Stochastic Differential Equation . . . 24
2.5 Parallelization of the Code . . . 28
2.6 Testing the Algorithm . . . 31

3 Analytical Solutions **35**
3.1 Exact Spin-wave Solution . . . 35
3.2 Linearized Equation of Motion . . . 37

4 Multipole Methods **49**
4.1 Introduction . . . 49
4.2 Barnes-Hut Algorithm . . . 51
4.3 Key Mapping . . . 56
4.4 Expansion of the Dipolar Field . . . 58
4.5 Translation of the Origin of a Multipole . . . 63
4.6 Timing and Accuracy . . . 64

5 Applications **67**
5.1 Magnetization Curve of a 10x10x10 Cubic System . . . 67
5.2 Relaxation Times of One Dimensional Systems . . . 71
5.3 Switching Times of a Nano-Cluster on a Surface . . . 76

A Appendix A **87**
A.1 Stochastic Integration Algorithm . . . 87

A.1.1 Translation of the stochastic differential equation 87
A.1.2 Runge-Kutta Integration Scheme 88
A.1.3 Pseudo-random number generator 89
A.2 Units Conversion 89
A.3 Amdahl's law 90
A.4 Domain Walls 91
A.5 Implementation of the Multipole Method 92
A.6 Dispersion relation 94
A.6.1 Two dimensional hexagonal lattice 94
A.6.2 Two dimensional tetragonal lattice 94

1 Introduction

One of the most important applications of magnetic properties of materials today is on mass storage in high tech devices. Magnetic discs are used in hard drives to store information by individually setting the magnetization direction of each block either 'up' or 'down'. The storage capacity has been increased dramatically reducing the size in each block which stores a bit. However, by reducing the size, its magnetization becomes more sensitive to thermal fluctuations which can lead to a superparamagnetic behavior and thus threaten the data integrity.

The technological developments and future challenges attract scientists to investigate confined magnetic nanostructures. The formation of FePt [1] and FeCo [2] nanoparticles ($d \approx$ 3-13nm) in a nanocrystal superlattice has been found. They might be used for future storage applications in which higher data storage densities[1] could be achievable. In recent years several groups have been able to depose and study one dimensional chains and clusters on surfaces. Gambardella et al [4] were able to investigate cobalt chains at steps of a platinum surface. A huge increase of the magnetocrystalline anisotropy was found compared to the bulk case. A similar behavior was observed by Brune et. al [5] which reports a 20 times higher anisotropy for cobalt islands on platinum compared to the bulk case. Other groups are investigating the switching process of hard magnetic materials due to magnetic fields. They try to reach the ultimate speed limit [6, 7] of magnetic switching by the use of a very short pulsed high magnetic field which is supposed to momentary collapse the ferromagnetic order.

These recent scientific developments were a motivation to study the time-dependent evaluation of magnetic properties of nano-structures on the atomic scale at finite temperature. The approach is based on a model Hamiltonian of the spin system, the so-called Heisenberg Hamiltonian. Thus, the aim of this thesis is a description of the time evolution of magnetic moments of nanostructures on the atomic scale at a finite temperature. Model parameters, e.g. exchange and anisotropy constants can be calculated using density functional theory (DFT) methods. By this atomistic treatment different lattice structures, anti-ferromagnets, interfaces or diluted magnetic semiconductors [8] can be described. A quantum mechanical treatment of the atomic spins would lead to an exponential[2] increase of the complexity which will be prohibitive for the treatment of large systems. A classical approximation is therefore used, where the quantum mechanical

[1] A data density several terabits/cm^2 could be reached. As a comparison modern hard drives have around 35 gigabits/cm^2 [3].

[2] The complexity will be of the order of $(2S+1)^N$ where N is the number of spins and S the spin quantum number.

spin operators will be substituted by classical angular momentum vectors.

The scope of the atomistic method is different from the one of continuum micromagnetic methods. Micromagnetic simulations are widely used to study large-scale systems under the assumption of a continuous magnetization, which is numerically treated by finite-difference or finite-element methods. However, the atomistic structure is not explicitly taken into account and their use is restriced to ferromagnetic materials.

The description of magnetic properties using the Heisenberg Hamiltonian at first requires a decoupling of the spin system to the lattice as well as the electronic system. There are several ways to reintroduce this coupling in a phenomenological and computationally tractable way. The most commonly used approaches are Monté Carlo methods which randomly distribute the states of each spin in such a way that a thermodynamical distribution is formed. Equilibrium properties like magnetization curves can suitably be described by using the Monté Carlo approach. However, the trajectory of each spin is totally unphysical and non-equilibrium time dependent properties cannot be described. There are methods known as kinetic Monté Carlo [9, 10], introducing a transition rate which relates each Monté Carlo step to a physically interpretable time. However, these methods assume a low propability of atomic spin switches in form of Poisson processes. Thus, correlation effects between neighboring spins, like spin-waves cannot be described, although in the limit of high switching times, good results can be achieved.

On the other hand, there exists the spin-dynamics approach, within which the exact trajectory of each spin is followed in time using a classical equation of motion, known as the Landau-Lifshitz equation. The spin system is coupled to an external system, the heat bath, to simulate a constant predefined temperature. It is introduced by adding a fluctuation field and a dissipative damping term to the classical equation of motion, which is presented in Chapter 2. The fluctuating term leads to a stochastic differential equation which is solved using a high order Runge-Kutta scheme for differential equations with weak noise.

A Zeeman term is also added to describe an external magnetic field acting on each spin. Besides, a term simulating the magnetocrystalline anisotropy is added describing the behavior of a spin to favor certain magnetization directions. This represents the interaction between the spin of an electron with its orbital momentum and is, therefore, called spin-orbit coupling. The energy scale is usually much smaller than the Heisenberg exchange. But it plays an important role for the direction of the magnetic moment since the Heisenberg model is isotropic.

In addition, the dipole-dipole interaction, which brings about the shape anisotropy, must be taken into account. Each atomic magnetic moment creates a dipolar magnetic field which couples to other atomic moments. The dipolar coupling between two magnetic moments is, for neighboring atoms, weak compared to the exchange coupling, but is important for large scaled systems since it is a long range interaction. However, this

also leads to a computational burden, since every spin is interacting with every other spin in the system. For N spins, this will lead to an increase of the computational time proportional to N^2. There are methods based on the evaluation of the multipole moment of regions, known as the Fast Multipole method and the Barnes and Hut tree algorithm, which can reduce the complexity to a $\mathcal{O}(N \log N)$ behavior. The downside is a controlable inaccuracy of the result. Based on the Barnes and Hut tree code, a program for the rapid evaluation of the dipolar field for a spin system has been developed, which is discussed in Chapter 4.

Apart from simulating the time evolution of the spin systems in an exact manner the classical equation of motion was studied analytically. Some exact spinwave solutions for an infinitely large system are presented in Chapter 4, which are used for testing the spin-dynamics code. Our interest mainly focuses on finite clusters or nanoislands. Based on the linearized equation of motion for the case of low excitations of ferromagnetic systems an eigenvalue equation was set up. The eigenmodes as well as the density of states for nanoislands on a substrate have been investigated and samples of different shaped nanoislands are discussed.

In Chapter 5, applications of the developed spin-dynamics code are presented. First of all, to ensure the correctness of the algorithm for equilibrium properties, results of the atomistic spin-dynamics code are compared to ones determined by a Monté-Carlo code. The magnetization curve as well as the susceptibility are in perfect agreement. Afterwards the demagnetization of infinitely long chains is investigated. The Mermin-Wagner theorem [11] predicts that there is no long range magnetic order in one-dimensional spin chains with short-ranged interactions. This theorem could be verified and the time could be determined in which the system loses its magnetic long-range order, depending on different parameters like damping parameter and temperature. Finally, the thermally-induced switching time of differently sized nanoislands deposed on a surface is studied. Different types of switching behaviors depending on the strength of the coupling constant are observed. For high values of the anisotropy constant, domains are propagating through the nanoisland with sharp domain walls whereas for low anisotropies a more collinear like switching is observed. It is also shown that the switching time follows the Arrhenius-Néel-law.

Conventions

This work tries to stick to the following conventions for mathematical formulas. The standard notation for vectors is to indicate them by bold face: $\boldsymbol{S}$. Its components will be given by a superscript index in Greek letters and if a set of vectors (i.e. spins) is given then a subscript index will denote each element: $S_i^\alpha, \quad \alpha = (x, y, z), i = (1, 2, 3, ...)$ Other definitions are given in the following table:

type	**style**	**example**	**exception**
scalar	standard	s	
vectors	bold	$\boldsymbol{v}$	
matrices	calligraphic	$\mathcal{M}$	$\mathcal{H}$ - Hamilton function $\mathcal{O}(x^n)$ - order of an expansion
units	roman	mRyd	i - imaginary unit number
operators	hat symbol	$\hat{S}_i$	
position index	Latin alphabet	$\boldsymbol{S_i}, \quad i = 1, ..N$	
direction index	Greek alphabet	$S^\alpha, \quad \alpha = x, y, z$	
gradient	bold denominator	$(\partial/\partial \boldsymbol{S})$	
set	curly brace	$\{\boldsymbol{S}\} = \boldsymbol{S}_1, \boldsymbol{S}_2, ...\boldsymbol{S}_N$	

2 Landau-Lifschitz Dynamics

2.1 Introduction

In this chapter the classical treatment of a quantum mechanical spin system based on a model Hamiltonian is introduced. The underlying assumption of this model is the formation of localized atomic moments, which are subject to an inter-atomic exchange interaction. This approximation is appropriate for systems with localized magnetic moments (like 4f transition elements). The interaction of the individual atomic magnetic moments is then described using a model Hamiltonian.

In analogy to the quantum mechanical equation of motion for the expectation value, a classical equation of motion describing the time evolution of the spin model is introduced. Also, a temperature bath is included by adding fluctuation and dissipation terms which simulate relaxation and excitation of magnons in solids. A parallelization scheme is presented which can be used for performing large scale calculations on supercomputers like Jugene at the Forschungszentrum Jülich. Finally, results of test systems are presented in order to verify the correctness and the accuracy of the code.

Length Scales

Magnetism plays a role on a wide range of length scales. Each electron carries a magnetic moment associated to its spin. The magnetic moment $\boldsymbol{\mu}$ of the spin $\boldsymbol{s}$ is coupled ($\boldsymbol{\mu} = \gamma \boldsymbol{s}$) to its spin by the gyromagnetic γ ratio which, for the electron, defined by the product of the Bohr magneton μ_B and the Landé factor g: $\gamma = -g\mu_B$. Each magnetic moment produces a dipolar field which can interact with other magnetic moments. However, the influence of the dipolar interaction between neighboring atoms is neglectable. The formation of the magnetic moments can be understood by the Coulomb interaction and the Pauli principle,which states that two electrons are not allowed to occupy the same state in an atom. An atomic moment can only be formed if the valence shell is not fully occupied. The valence shell will first be filled up with spins of the same state (i.e. up or down). This effect is expressed by the first Hund's rule. Therefore, an atom will have a maximum spin moment if the valence shell is half filled. The strength of an interaction depends on the overlap of the wave functions of the electrons. Therefore, the interaction among the electrons of an atom is orders of magnitude higher than the interaction between electrons of different atoms. Thus, for the study of magnetic properties, it is a good approximation to treat the electrons of one atom as a single atomic macrospin which is interacting with its surroundings.

The inter-atomic exchange coupling between atomic moments also stems from an interplay between the Coulomb interaction and the Pauli principle as well. This leads to the well known Heisenberg exchange Hamiltonian which is governing the interaction of nearby atoms. In addition, spin space and real space are coupled via the spin-orbit interaction, which is partly responsible for the spatial directionality of the magnetization. The emerging anisotropy is called magnetocrystalline anisotropy.

By increasing the system size above the nanometer scale the long range dipole-dipole interaction becomes more and more important. The dipole-dipole interaction is the second term which is responsible for the spatial orientation of the magnetic moments and is competing with the magnetocrystalline anisotropy. Unlike the magnetocrystalline anisotropy the energy minimum of the dipole-dipole interaction is mainly determined by the shape of the sample. The energy minimum of the full Hamiltonian, consisting of the Heisenberg, the magnetocrystalline Anisotropy and the dipolar interaction, and leads to a formation of domains, while the competition between the Heisenberg exchange and the magnetocrystalline anisotropy determines the thickness of domain walls.

2.2 Hamiltonian

2.2.1 Heisenberg Hamiltonian

The ordering of the magnetic moments in solids is mainly driven by the Heisenberg exchange interaction. This interaction cannot be understood within the single-electron approximation which is often used in solid state physics. Two effects have to be included to understand the exchange interaction. The Pauli principle and the Coulomb interaction between electrons.

According to the Pauli principle the many-body wave function must be anti-symmetric for a permutation of two identical fermions. One consequence is that two identical electrons cannot occupy the exact same position. Hence, the energy due to the Coulomb interaction between them is lowered, compared to two electrons with a different spin state in the same electronic state.
This phenomenon can quantitatively be understood studying a hydrogen molecule. The two-electron wave function consists of a spatial and a spin part:

$$\Psi(\boldsymbol{r_1}, \boldsymbol{r_2}, s_1, s_2) = \psi(\boldsymbol{r_1}, \boldsymbol{r_2}) \cdot \chi(s_1, s_2),$$

$\boldsymbol{r}$ defines the position and s the spin of the electron. The single spin eigenfunctions can be combined to eigenfunctions of the total spin operator, which is for two electrons known as the anti-symmetric singlet state and the symmetric triplet state. Hence, the spatial part needs to be symmetric for the singlet state and anti-symmetric for the triplet state.

The Hamiltonian of an hydrogen molecule is given by:

$$\mathcal{H} = \mathcal{H}(\boldsymbol{r}_1) + \mathcal{H}(\boldsymbol{r}_2) + \mathcal{H}_{e-e}(\boldsymbol{r}_1, \boldsymbol{r}_2)$$

where $\mathcal{H}(\boldsymbol{r_1})$ and $\mathcal{H}(\boldsymbol{r_2})$ denote the single electron Hamiltonian of the hydrogen molecule using $\boldsymbol{r}_1$ and $\boldsymbol{r}_2$ which defines the position of the electrons. $\mathcal{H}_{e-e}(\boldsymbol{r_1}, \boldsymbol{r_2})$ denotes the Coulomb exchange interaction between the electrons. A first approximation of the solution of the Hamiltonian above is given by an ansatz of products of the one electron wave functions. In the Heitler-London approximation the wave functions are given by:

$$\psi_\pm(\boldsymbol{r_1}, \boldsymbol{r_2}) = \phi_A(\boldsymbol{r_1})\phi_B(\boldsymbol{r_2}) \pm \phi_B(\boldsymbol{r_1})\phi_A(\boldsymbol{r_2})$$

Here, ϕ_A and ϕ_B denotes the atomic wave function of an electron around a nucleus A or B with the coordinates $\boldsymbol{r}_A$, $\boldsymbol{r}_B$. The '+' wave function corresponds to the singlet, the '−' corresponds to the triplet spin solution. The energy of the two wave functions can easily be determined [12]:

$$E = \frac{\langle\psi(\boldsymbol{r_1}, \boldsymbol{r_2})|\mathcal{H}|\psi(\boldsymbol{r_1}, \boldsymbol{r_2})\rangle}{\langle\psi(\boldsymbol{r_1}, \boldsymbol{r_2})|\psi(\boldsymbol{r_1}, \boldsymbol{r_2})\rangle} = 2E_I + \frac{C \pm A}{1 \pm S} \tag{2.1}$$

where the following integrals appear and need to be evaluated:

$$E_I = \int \phi_A^*(\boldsymbol{r_1})(-\frac{\hbar^2}{2m}\boldsymbol{\nabla}_1^2 - \frac{e^2}{4\pi\epsilon_0 r_{A1}})\phi_A(\boldsymbol{r_1})d\boldsymbol{r}_1 \tag{2.2}$$

$$C = \frac{e^2}{4\pi\epsilon_0}\int(\frac{1}{r_{AB}} + \frac{1}{r_{12}} - \frac{1}{r_{A2}} - \frac{1}{r_{B1}})|\phi_A(\boldsymbol{r_1})|^2|\phi_B(\boldsymbol{r_2})|^2 d\boldsymbol{r}_1 d\boldsymbol{r}_2 \tag{2.3}$$

$$A = \frac{e^2}{4\pi\epsilon_0}\int(\frac{1}{r_{AB}} + \frac{1}{r_{12}} - \frac{1}{r_{A1}} - \frac{1}{r_{B2}})\phi_A^*(\boldsymbol{r_1})\phi_A(\boldsymbol{r_2})\phi_B(\boldsymbol{r_1})\phi_B^*(\boldsymbol{r_2})d\boldsymbol{r}_1 d\boldsymbol{r}_2 \tag{2.4}$$

$$S = \int \phi_A^*(\boldsymbol{r_1})\phi_A(\boldsymbol{r_2})\phi_B(\boldsymbol{r_1})\phi_B^*(\boldsymbol{r_2})d\boldsymbol{r}_1 d\boldsymbol{r}_2 \tag{2.5}$$

The symbol r_{A1} is denotes the distance between the coordinates $\boldsymbol{r}_A$ and $\boldsymbol{r}_1$. The previous integrals are known as the ionization energy of a hydrogen atom E_I,the Coulomb integral C,the exchange integral A and the overlap integral of the wave functions S. The spatial part of the singlet wave function needs to be symmetric (ψ_+), the spatial part of the triplet wave function anti-symmetric (ψ_-), which leads to two different eigenenergies, E_s for the singlet and E_t for the triplet state. Their difference will be denoted as the coupling constant J:

$$J := E_s - E_t = -\frac{CS - A}{1 - S^2}$$

It can be either negative or positive which leads to a ferromagnetic or an anti-ferromagnetic coupling. The splitting of the four degenerate spin states will be small compared to other excitation energies. As the interest focuses on the spin properties, it is useful to work with a Hilbert space consisting of the four spin states. The following Hamiltonian fulfills these properties and can also reproduce the eigenvalues of the original Hamiltonian:

$$\mathcal{H}_{\text{spin}} = \frac{1}{4}(E_s + 3E_t) - (E_s - E_t)\hat{\boldsymbol{S}}_1 \cdot \hat{\boldsymbol{S}}_2$$

The operator will have an eigenvalue of E_t in the triplet and E_s in the singlet state. The first part of the sum is constant and, therefore, does not change the physical behavior. Therefore, this part is neglected and one writes the spin Hamiltonian in the following form:

$$\mathcal{H} = -J\hat{\boldsymbol{S}}_1 \cdot \hat{\boldsymbol{S}}_2$$

The previous term was first introduced by Heisenberg in 1920 and is called the Heisenberg Hamiltonian. In principle, coupling constants between every spin pair in the system has to be considered. Nevertheless, since the overlap of the wave functions decays exponentially with the atomic distance only nearby atoms have a finite coupling constant. Therefore, in the simulations presented in the next chapters, we restrict ourselves to a nearest neighbor treatment. A positive coupling constant leads to a ferromagnetic coupling, a negative one to an antiferromagnetic coupling. The Heisenberg model provides a good description for localized states i.e. 4f transition elements. It is a less good approximation in the description of 3d transition elements where a model for itinerant electrons will be more applicable.

The classical Heisenberg Hamiltonian is given by a substitution of the spin operators by c-numbers[1]. In the limit of high temperature and high spin quantum numbers the difference between quantum and classical spins becomes less important leading to the applicability of the classical model:

$$\mathcal{H} = -J\boldsymbol{S}_1 \cdot \boldsymbol{S}_2$$

The length of the classical spin correspond to $|\boldsymbol{S}| = \sqrt{s(s+1)}$ where s is the spin quantum number.

2.2.2 Zeeman Term

A system of spins can interact with an external magnetic field $\boldsymbol{H}_{\text{Zeeman}}$. This will lead to a precession of the spin around the magnetic field. The spin motion of a single electron can be described using the Zeeman Hamiltonian:

$$\mathcal{H}_{\text{Zeeman}} = -\boldsymbol{\mu} \cdot \boldsymbol{H}_{\text{Zeeman}} = g\mu_B \boldsymbol{H}_{\text{Zeeman}} \cdot \boldsymbol{S} \tag{2.6}$$

At the energy minimum, the spin lies antiparallel to the magnetic field, as the electron magnetic moment is antiparallel to the electron spin.

2.2.3 Magnetocrystalline Anisotropy

The Heisenberg model mentioned above depends only on the spin states in the system and its coupling constants. Thus, the system is invariant to a collective rotation of the spins. In real systems this behavior is not observable. Due to the shape of the sample

[1] Nomenclature introduced by Paul Dirac to distinguish between real or complex numbers (c-numbers) and operators.

and its crystalline structure the spins in a system favor certain direction, which is called the *easy axis* (or they can favor the alignment in a plane which is denoted as the *easy plane*). In particular, in systems of reduced dimension and finite size[2], there is a high contribution of the magnetocrystalline anisotropy caused by the spin-orbit interaction of the electrons .

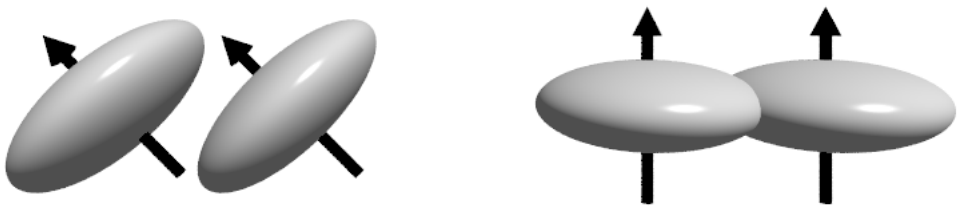


Figure 2.1: Schematic description of the deformation of the atomic orbitals of two electrons depending on the spin direction. The overlap between two orbitals changes, leading to a change of the total energy. Hence, certain spin states are energetically preferable.

Roughly speaking, the spin of the electron interacts with the angular momentum of the atomic orbitals. Depending on the direction of the spin there is a small variation of its atomic orbital (see 2.1), which changes the overlap of the wave function with its neighbors.
Quantitatively, starting from the Dirac equation one can derive [13] an additional perturbation term in the Schrödinger equation. For a spherical potential one gets:

$$\left(-\frac{\hbar^2}{2m}\boldsymbol{\nabla}^2 + V(\boldsymbol{r}) + \frac{\mu_B}{2mc}\underbrace{\frac{1}{r}\frac{\partial V(r)}{\partial r}}_{\xi}(\boldsymbol{\sigma}\cdot\boldsymbol{L}) \right)\Psi = E\Psi \tag{2.7}$$

Here, $\boldsymbol{\sigma}$ is a vector containing the Pauli matrices. The strength of the additional term in equation (2.7) depends on the angular momentum $\boldsymbol{L}$ of each orbital and a prefactor ξ, which is influenced by the nucleus potential $V(r)$. Hence, one can expect a high anisotropy constant for heavy atoms. But it also highly depends on the shape of each orbital. Assuming a Coulomb potential, the factor ξ can be derived to $\xi = \frac{1}{r}\frac{\partial dV(r)}{\partial r} \sim \frac{1}{r^3}$. Thus, the contribution to ξ is strong at r close to zero. The s-orbitals do not contribute to the magnetocrystalline anisotropy because they do not carry an angular momentum. As a rule of thumb, one can say that the contribution is decreasing from the p-orbitals (most closely located around the nucleus) to the d- and f- orbitals.

[2]Clusters, chains and ad-atoms.

A first-order perturbation term can be derived which is used in the classical spin-dynamics simulation:

$$H_{ani} = -K \cdot (\boldsymbol{n} \cdot \boldsymbol{S})^2$$

Here, $\boldsymbol{n}$ is a normalized vector pointing to the easy axis (or the orientation vector of an easy plane) and K is known as the anisotropy constant. The anisotropy constant is usually much smaller than the coupling constant (typically $10^{-3}J - 10^{-1}J$). For positive values of K the system tries to align the spins parallel or anti-parallel to the vector $\boldsymbol{n}$. For negative values, the spins align perpendicular to $\boldsymbol{n}$. Such situations are often met in thin films where the translational symmetry is broken. For symmetry reasons, in thin films, the vector $\boldsymbol{n}$ is pointing perpendicular to the surface of the plane. Depending on the sign of K the atomic moments align parallel (in- plane magnetization) or perpendicular (out-of-plane magnetization) to the surface. In recent years hard disc manufacturers have been able to use an out-of-plane magnetization instead of an in-plane magnetization to further extend storage capacities in hard discs.

2.2.4 Dipole-Dipole-Interaction

The dipolar interaction between two atoms is about three magnitudes smaller than the exchange coupling [14]. However, while the latter drops exponentially, the dipolar interaction is of long range nature, decaying proportionally by $1/r^3$ with the distance r.. Hence, while it almost does not influence the behavior of the inter-atomic interaction between neighboring atoms, it crucially influences the magnetic properties of large clusters. The dipolar interaction basically induces two effects: the magnetic shape anisotropy [15] and the formation of domains. According to classical magnetostatics, the field induced by a magnetic moment $\boldsymbol{\mu}$ is given by:

$$\boldsymbol{B}(\boldsymbol{r}) = \frac{3\boldsymbol{r}(\boldsymbol{r} \cdot \boldsymbol{\mu}) - \boldsymbol{\mu} \cdot r^2}{r^5}$$

Another magnetic moment, which is present in the system will couple to the magnetic field produced by a $\boldsymbol{\mu}$. Using the Zeeman equation (2.6) the dipole-dipole energy of a system of magnetic moments $\{\boldsymbol{\mu}\}$ at position $\{\boldsymbol{r}\}$ is given by:

$$\mathcal{H}_{dip}(\boldsymbol{r}) = \frac{1}{2} \sum_{i \neq j} \frac{\boldsymbol{\mu}_i \cdot \boldsymbol{\mu}_j (\boldsymbol{r}_i - \boldsymbol{r}_j)^2 - 3(\boldsymbol{\mu}_i \cdot (\boldsymbol{r}_i - \boldsymbol{r}_j))(\boldsymbol{\mu}_j \cdot (\boldsymbol{r}_i - \boldsymbol{r}_j))}{|\boldsymbol{r}_i - \boldsymbol{r}_j|^5},$$

where i and j define the atoms between which the interaction is calculated. This equation can be simplified [16] for ferromagnetic systems with a uniform distribution of magnetic moments leading to the energy:

$$E = \frac{1}{2}|\boldsymbol{\mu}|^2 \sum_{i \neq j} \frac{1 - 3\cos^2 \theta_{ij}}{R_{ij}^3}$$

where R_{ij} is the distance between spin i and spin j and θ_{ij} is the angle between the joining vector of spin i and spin j and their magnetic moment. For a chain of magnetic moments, the angle θ_{ij} will be independent of the atomic positions. Hence, it

is easy to see that the dipolar energy is minimized if the magnetic moments align in the direction of the chain ($\theta_{ij} = 0$). However, for an infinite anti-ferromagnetic coupled chain of magnetic moments, the spins will align perpendicular to the chain direction [17].

The minimum of the dipolar energy for infinite systems depends on the dimension of the system. For an ordered ferromagnetic two dimensional system the magnetization direction will also be in plane. This is in contradiction to the easy axis of a magnetocrystalline anisotropy which in many cases points out of plane. Thus, a competition between the two terms will determine the ground state. It is worth mentioning that the dipole-dipole energy does not affect the ground state of an infinite three dimensional bulk system. The dipolar energy is purely determined by the boundaries of the sample. Therefore, in a finite system the ground state for the dipolar energy will be determined by the shape of the sample. For this reason,the orientation of the ground state due to the dipole-dipole interaction is often called shape anisotropy.

A ferromagnetic state for a finite particle is often very inefficient in terms of dipolar interactions. Indeed, the dipolar energy is reduced by the creation of domains with different magnetic orientations These domains are separated by domain walls with different thicknesses.

Such a magnetic subdivision is, of course, payed by the exchange energy which is lowered when all spins are pointing towards the same direction. However, since the exchange interactions are short ranged, the exchange energy only increases at the boundaries between domains, the domain walls. In fact, the exchange interactions broaden the width of the domain wall to make the transition between one magnetic domain to the closest one as smooth as possible in contradiction to the magnetocrystalline anisotropy, which tends to decrease the domain wall. Hence, a competition between the magnetocrystalline anisotropy and the exchange energy will determine the domain wall width as derived in Appendix A.4.

In calculations, the long range dipole-dipole interaction leads to a computational burden. The exchange energy is a local interaction. Therefore, the number of interaction partners does not increase with the number of atoms N, whereas the number of magnetic dipolar interactions increases linearly. This leads to a increase of the computational time which is proportional to the number of moments squared: N^2. For the sake of comparison, a system with a local coupling leads to a linear computational increase. As discussed in Chapter 5, strategies can be found facilitating the computational time to an (almost) linear behavior.

2.3 Classical Approach

In this section, the quantum mechanical equation of motion for a Heisenberg spin system is introduced. Then a classical analogon [18] will be derived, which is extended by

introducing a coupling to a heat bath. Therefore, a Langevin-like approach is used introducing fluctuating and dissipating terms to the equation of motion.

2.3.1 Classical Equation of Motion

Motion of the Angular Momentum of a Charge Distribution

The spin of an electron can classically be interpreted as a rotation of a finite sized particle around one of its symmetry axes. One considers that the electron charge is distributed in a finite volume according to a charge density distribution $\rho(\boldsymbol{r})$. If one applies a magnetic field $\boldsymbol{B}$ a Lorenz force will be induced on the particle, leading to a precession.

The derivative of the angular momentum is given by the torque, $\boldsymbol{\mu} \times \boldsymbol{B}$. Hence, one ends up with the equation of motion $\partial \boldsymbol{L}/\partial t = \boldsymbol{\mu} \times \boldsymbol{B}$ [19]. Here we recall once more that for a spin of an electron, the angular momentum $\hbar \boldsymbol{S}$ is connected to the magnetic moment $\boldsymbol{\mu}$ by the gyromagnetic ratio $\boldsymbol{\mu_e} = -g\mu_B \boldsymbol{S}$ where $\mu_B = \frac{|e|\hbar}{2mc}$ is the Bohr magneton and g refers to the Landé factor. The precession of a spin in a magnetic field can be interpreted in a classical picture, where the spin is viewed as a classical angular momentum:

$$\frac{\partial \boldsymbol{S}}{\partial t} = -\frac{g\mu_B}{\hbar} \boldsymbol{S} \times \boldsymbol{B}$$

According to the equation of motion the spin will precess around the magnetic field as seen in Fig. 2.2. Comparing the Heisenberg Hamiltonian ($\mathcal{H} = -J\boldsymbol{S} \cdot \boldsymbol{S}'$) with the Zeeman Hamiltonian ($\mathcal{H}_{\text{Zeeman}} = g\mu_B \boldsymbol{S} \cdot \boldsymbol{H}_{\textbf{Zeeman}}$) one can identify the neighboring spin $\boldsymbol{S}'$ with an effective magnetic field which acts on the spin $\boldsymbol{S}$:

$$g\mu_B \boldsymbol{H} \equiv -J\boldsymbol{S}'$$

Hence, the equation of motion for two atomic spins coupled by an Heisenberg Hamiltonian is defined as :

$$\frac{\partial \boldsymbol{S}}{\partial t} = -\frac{J}{\hbar} \boldsymbol{S}' \times \boldsymbol{S}$$

An analogous equation of motion is also valid for spin $\boldsymbol{S}'$, which leads to a system of coupled ordinary differential equations. This derivation is up to now just a rough estimation of the motion of a classical spin. A more general equation of motion can be motivated by finding an analogon for the quantum mechanical commutator relation.

Quantum Mechanical Spin Dynamics

The quantum-mechanical spin operators at atoms i,j obey the following commutator relations:

$$[\widehat{S}_i^{\alpha}, \widehat{S}_j^{\beta}] = i\hbar\delta_{ij}\epsilon_{\alpha\beta\gamma}\widehat{S}_i^{\gamma}, \quad \alpha, \beta, \gamma = (x, y, z) \quad i, j = 1, ..N \tag{2.8}$$

where $\epsilon_{\alpha\beta\gamma}$ denotes the fully antisymmetric Levi-Civita-tensor and δ_{ij} is the Kronecker's delta. The commutator bracket is defined as $[A, B] = AB - BA$. For an arbitrary observable $\widehat{O}$ the quantum equation of motion in the Heisenberg picture is determined

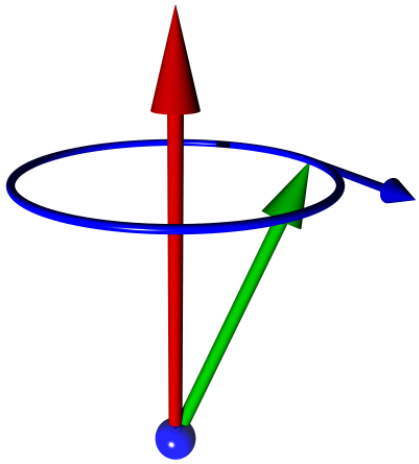


Figure 2.2: Motion of a spin (green) around an effective magnetic field (red). The trajectory (blue) shows the precession around the magnetic field.

by $i\hbar\frac{\partial\widehat{O}}{\partial t} = [\widehat{O}, \mathcal{H}]$. Hence, for the spin operators the quantum mechanical equation of motion is given by:

$$i\hbar\frac{\partial\widehat{S}_i^\alpha}{\partial t} = [\widehat{S}_i^\alpha, \mathcal{H}(\{\widehat{\boldsymbol{S}}\})], \quad \alpha,\beta = (x,y,z) \quad i,j = 1,..N \tag{2.9}$$

Classical Spindynamics

In the classical approximation the quantum mechanical operators are substituted by c-numbers:

$$\widehat{\boldsymbol{S}} \rightarrow \boldsymbol{S}$$

An analogon to the quantum mechanical commutator in classical mechanics for spin systems is given by the Lie-Poisson bracket [20], which for spin systems reads:

$$\{F(\{\boldsymbol{S}\}), G(\{\boldsymbol{S}\})\} = \sum_{i=1}^{N} \epsilon_{\alpha\beta\gamma}\frac{\partial F}{\partial S_i^\alpha}\frac{\partial G}{\partial S_i^\beta}S_i^\gamma \tag{2.10}$$

where $F(\{\boldsymbol{S}\})$ and $G(\{\boldsymbol{S}\})$ are functions of the set of vectors $\{\boldsymbol{S}\}$. Using $F(\{\boldsymbol{S}\}) = S_i^\alpha$ and $G(\{\boldsymbol{S}\}) = S_j^\beta$ in function (2.10) the following relation can be derived:

$$\{S_i^\alpha, S_j^\beta\} = \epsilon_{\alpha\beta\gamma}\delta_{ij}S_i^\gamma \tag{2.11}$$

This equation can be interpreted as the classical analogon to the quantum mechanical spin commutator relationship (2.8). Thus, for an observable, the time evolution is given

by the Lie-Poisson bracket (in analogy to equation (2.9)):

$$\hbar \frac{\partial O}{\partial t} = \{O, \mathcal{H}(\{\boldsymbol{S}\})\} \tag{2.12}$$

For the spin component α of the spin $\boldsymbol{S}$ at position i the equation of motion is given by

$$\begin{aligned} \hbar \frac{\partial S_i^\alpha}{\partial t} &= \{S_i^\alpha, \mathcal{H}(\{\boldsymbol{S}\})\} \\ &= \sum_{j=1}^{N} \epsilon_{\alpha\beta\gamma} \delta_{ij} \frac{\partial \mathcal{H}}{\partial S_j^\beta} S_j^\gamma \\ &= (\frac{\partial \mathcal{H}}{\partial \boldsymbol{S}_i} \times \boldsymbol{S}_i)^\alpha \,, \end{aligned}$$

which leads to

$$\hbar \frac{\partial \boldsymbol{S}_i}{\partial t} = (\frac{\partial \mathcal{H}}{\partial \boldsymbol{S}_i} \times \boldsymbol{S}_i) \tag{2.13}$$

Equation (2.13) is the classical equation of motion for the spin Hamiltonian used in this work. It is a first-order differential equation for which the spin directions in space are sufficient as an initial condition for solving the equation of motion. The derivative of the Hamiltonian $\partial \mathcal{H} / \partial \boldsymbol{S}_i$ will be denoted as the local field splitting or the effective field:

$$\boldsymbol{H}_{eff} = \partial \mathcal{H} / \partial \boldsymbol{S}_i$$

It can be interpreted as a local, effective magnetic field which consists of the sum of all interactions acting on the particular spin. The cross product in the equation of motion leads to a precession of the spin around its local field (see Fig. 2.2).

Using relation (2.12) one can easily show that the energy is conserved using the equation of motion (2.13) by a calculation of $\frac{\partial \mathcal{H}}{\partial t} = \{\mathcal{H}(\boldsymbol{S}), \mathcal{H}(\boldsymbol{S})\} = 0$ as well as the conversation of the length of each single spin $\boldsymbol{S}_i$ by determining $\partial \boldsymbol{S}_i^2 / \partial t = \{(S_i^\alpha)^2 + (S_i^\beta)^2 + (S_i^\gamma)^2, \mathcal{H}(\boldsymbol{S})\} = 0$. The conservation of the length of each spin will be used later on to determine the numerical error of the integration scheme.

2.3.2 Simulation of Damping

A solution of the equation of motion (2.13) which was derived before would lead, in the case of a single spin in an external field, to a periodic precession of the spin around a magnetic field at a constant energy because the spin system is totally decoupled from the real space. This behavior would be unphysical, especially for systems at low temperature. Here, one would expect a tendency of the spin to lose its energy (for zero temperature), which would lead to a relaxation of the spin to the ground state. This phenomenon is mainly driven by interactions with the crystal lattice or the electronic

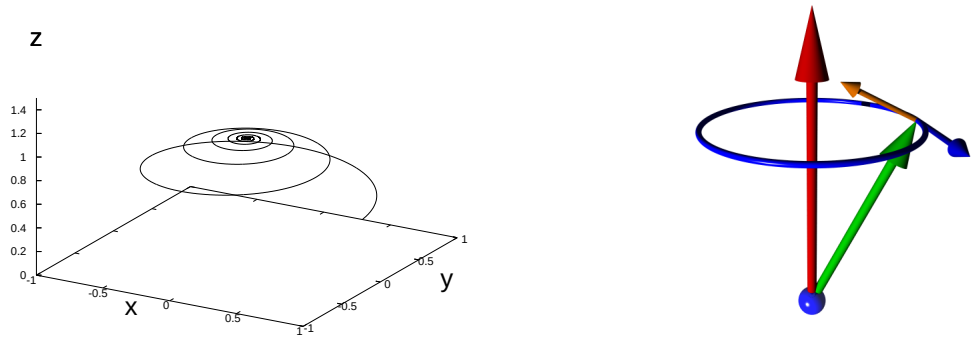


Figure 2.3: (left) Trajectory of the magnetic moment of a spin precessing around a magnetic field which is pointing in the z-direction. The system is damped ($\lambda, \alpha = 0.1$) with the Landau-Lifschitz or the Gilbert damping. The trajectory is exactly the same. Just the timescale of the process is different. (right) Spin (green) processing around a local field (red) by a force (blue) driven by the Hamiltonian. The system is slowly damped by Landau-Lifschitz or Gilbert damping (orange).

structure. By decoupling the spin system these interactions (when assuming the Zeeman or Heisenberg Hamiltonian) have been excluded. However, the relaxations can be simulated introducing a coupling of the spin system by using model damping terms. In 1935 Landau and Lifschitz first introduced an additional term to the equation of motion to describe damping [21]:

$$\frac{\partial \boldsymbol{S}_i}{\partial t} = \frac{1}{\hbar}\boldsymbol{H}_{eff} \times \boldsymbol{S}_i - \lambda\frac{1}{\hbar}(\boldsymbol{H}_{eff} \times \boldsymbol{S}_i) \times \boldsymbol{S}_i \tag{2.14}$$

The strength of the damping is described by a positive damping constant λ. The damping force is by construction always directed toward the local energy minimum which can be seen in Fig 2.3.
Later Gilbert [22] proposed a different damping term which is proportional to the 'speed' of the spin, i.e. to the precession frequency:

$$\frac{\partial \boldsymbol{S}_i}{\partial t} = \frac{1}{\hbar}\boldsymbol{H}_{eff} \times \boldsymbol{S}_i + \alpha\boldsymbol{S} \times \frac{\partial \boldsymbol{S}}{\partial t} \tag{2.15}$$

At first sight these two equations look different. But it can be shown that they are mathematically equivalent by taking the cross product of equation (2.15) with $\boldsymbol{S}$:

$$\begin{aligned}
\boldsymbol{S} \times \frac{\partial \boldsymbol{S}}{\partial t} &= -\frac{1}{\hbar}(\boldsymbol{H}_{eff} \times \boldsymbol{S}) \times \boldsymbol{S} + \alpha\boldsymbol{S} \times (\boldsymbol{S} \times \frac{\partial \boldsymbol{S}}{\partial t}) \\
&= -\frac{1}{\hbar}(\boldsymbol{H}_{eff} \times \boldsymbol{S}) \times \boldsymbol{S} + \alpha(\boldsymbol{S}(\underbrace{\boldsymbol{S}\frac{\partial \boldsymbol{S}}{\partial t}}_{=0}) - \frac{\partial \boldsymbol{S}}{\partial t}S^2) \\
&= -\frac{1}{\hbar}(\boldsymbol{H}_{eff} \times \boldsymbol{S}) \times \boldsymbol{S} - \alpha S^2\frac{\partial \boldsymbol{S}}{\partial t}
\end{aligned} \tag{2.16}$$

This term can now be inserted into equation (2.15). A straight-forward calculation leads to:

$$(1 + S^2\alpha^2)\frac{\partial \boldsymbol{S}}{\partial t} = \frac{1}{\hbar}\boldsymbol{H}_{eff} \times \boldsymbol{S} - \frac{\alpha}{\hbar}(\boldsymbol{H}_{eff} \times \boldsymbol{S}) \times \boldsymbol{S} \tag{2.17}$$

By introducing a rescaling of time of the form $\tau = (1 + S^2\alpha^2)t$ the damping terms of equation 2.14 and equation 2.15 are mathematically equivalent. The trajectories are the same, however, the timescale on which the spin is 'moving' is different. As a result, this renormalization of the time leads to a crucial difference between the damping types in the high damping limit:

$$\begin{aligned} \lambda \to \infty &\implies \frac{\partial \boldsymbol{S}}{\partial t} \to \infty \quad (\text{Landau} - \text{Lifschitz}) \\ \alpha \to \infty &\implies \frac{\partial \boldsymbol{S}}{\partial t} \to 0 \quad (\text{Gilbert}) \end{aligned}$$

It is often argued that due to this behavior the Gilbert damping is preferable to use in simulations. Indeed, the Gilbert damping term is, as it is proportional to the spin velocity $\partial \boldsymbol{S}/\partial t$, similar to common damping terms known in classical mechanics. Hence, one would expect the spin to slow down with increasing damping parameter which is described by equation (2.18).

One could also interpret the damping term as an energy loss term which seems, in some cases, more reasonable for a system of atomic spins interacting with each other. Hence, the prefactors would be proportional to the energy loss of the system in time. Increasing it should lead to a faster relaxation of the spins towards its groundstate. Therefore, the Landau-Lifschitz interpretation could in some cases also be preferable.

In the low-damping limit both damping terms converge to the same result. Since mostly low-damped systems will be studied, the choice between Landau-Lifschitz and Gilbert damping is arbitrary. For higher damping values ($\lambda > 0.1$) it will explicitly be mentioned which damping term is used. The damping term is an important parameter in the simulations. For micromagnetic simulations, values around 0.01 [23, 24] are used in the literature which are often extracted from ferromagnetic resonance (FMR) experiments. In this thesis, several values have been used ranging from $\lambda = 0.001$ to 0.1.

2.4 Temperature Bath

As discussed, by introducing the Heisenberg model Hamiltonian, the spin system was decoupled from the lattice and the electronic system. Hence, no interaction between them will be described using the Landau-Lifshitz model. There are several approaches to reintroduce a coupling to simulate a temperature bath. The most prominent approach is the Monte Carlo method, which is suitable to describe equilibrium properties. However, time dependent non-equilibrium properties can hardly be determined. There are

approaches known as Kinetic Monte Carlo methods [9] which introduce a time leading to good results in the distribution but fail in the regime of ultrashort timescales. Other methods introduce the coupling of the spin and the lattice system via the exchange constant which then will depend on the oscillating distance between the lattice atoms [25, 26].

2.4.1 Langevin-Like Approach

In this thesis a Langevin-Like approach similar to molecular dynamics is used in which a thermal heat bath is acting on the system via stochastic forces. For an easy understanding it will first be introduced in the molecular dynamics context.

Brownian Motion

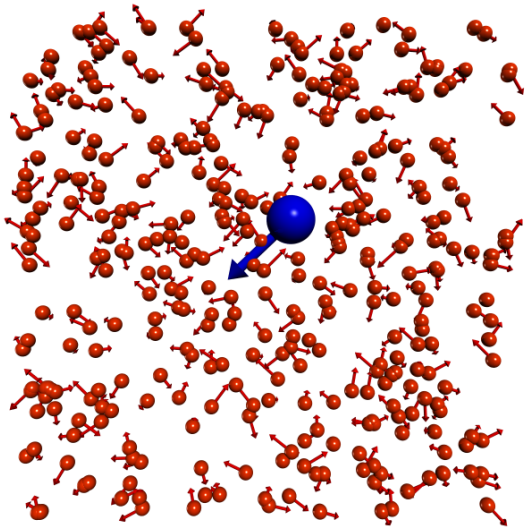


Figure 2.4: Motion of a particle (blue) in a liquid (red). The arrows show the direction of motion for the particles.

The motion of a particle in a liquid was first studied by Robert Brown and is, therefore, called a Brownian motion. Einstein [27] related the motion of a particle in a liquid to a stochastic model in which the liquid atoms are rapidly interacting with the particle (see Fig. 2.4). The motion is described by the Langevin equation which consists of Newton's second law including two additional terms:

$$m \cdot \frac{\partial^2 x}{\partial t^2} = F_{det}(x,t) - \lambda \cdot \frac{\partial x}{\partial t} + F_{sto}(x,t) \tag{2.18}$$

The first part on the right-hand side corresponds to deterministic forces which are known (or can be calculated) from the state of the system. The second term is a damping term which is proportional to the velocity of the particle acting as a friction term which

tends to bring the particle to stillstand. The last term, F_{sto}, is a stochastic force which accelerates the particle, mimicking the random, temperature-induced collisions of the particle with molecules of the liquid.

Stochastic Landau-Lifschitz equation

The previous method can also be applied to a system of classical spins. The classical spin will fluctuate due to excitations caused by interaction with i.e. phonons or electrons. Friction-driven energy loss will be described by the damping term introduced earlier. The equilibrium state is determined by an interplay between both mechanisms.

We will now focus on the description of thermal excitations of the spin system. These effects are simulated by including the following term in the equation of motion:

$$\boldsymbol{f}_i \times \boldsymbol{S} \tag{2.19}$$

Here $\boldsymbol{f}_i$ denotes a fluctuating term simulated by a stochastic distribution which can also be interpreted as an effective fluctuating magnetic field. The resulting motion vector is perpendicular to the spin, which thus conserves its length.

A large number of processes is responsible for spin excitations. These excitations can have different stochastic distributions. Using the central limit theorem, the sum of the different distributions will lead to a Gaussian distribution. It is also plausible that a thermal heat bath should be rotationally invariant and should not favoring any direction. In other words the thermal heat bath should not lead to a drift of the system. Hence, the mean value of the distribution must vanish:

$$\langle f_i^\alpha \rangle = 0$$

It will also be assumed that the fluctuations are uncorrelated for each spin as well as for each component of the spin. The fluctuating force should also be uncorrelated in time for each spin. This assumption is valid if the time interval $(t - t')$ between two subsequent stochastic collisions is short compared to the timescale of the spin motion and can mathematically be formulated as:

$$\langle f_i^\alpha(t), f_j^\beta(t') \rangle = \delta_{ij}\delta_{\alpha\beta}\delta(t - t')\epsilon^2$$

The Fourier transformation of this expression reads:

$$F(\omega) = \int_{-\infty}^{\infty} dt \langle f_i^\alpha(0), f_i^\alpha(t) \rangle \exp(-\mathrm{i}\omega t) = \epsilon^2 \tag{2.20}$$

Hence, the spectral density is independent of ω. The distribution will therefore be called white noise. The strength of the fluctuating force and the damping parameter should be chosen in a way that the distribution of the energy of a single particle in an equilibrium state follows a Boltzmann distribution:

$$P(E) \sim \exp(-\, E/k_B T)$$

This can be guaranteed by the following relation (Fluctuation-Dissipation-Theorem) between the temperature T, the damping constant λ and the standard deviation of the fluctuating force distribution:

$$\epsilon^2 = 2\lambda k_B T$$

A combination of motion, damping and fluctuation term leads to the final equation of

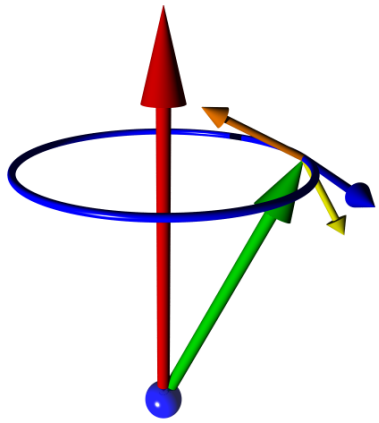


Figure 2.5: Visualization of the forces acting on a spin (green) precessing (blue) around a local field (red). The damping vector is colored in orange and one example for a fluctuating force vector is colored in yellow. All three vectors are perpendicular to the spin vector (green).

motion which is used for simulations in this work:

$$\frac{\partial \boldsymbol{S}_i}{\partial t} = \frac{\partial \mathcal{H}}{\partial \boldsymbol{S}_i} \times \boldsymbol{S}_i - \lambda(\frac{\partial \mathcal{H}}{\partial \boldsymbol{S}_i} \times \boldsymbol{S}_i) \times \boldsymbol{S}_i + \boldsymbol{f}(t) \times \boldsymbol{S} \tag{2.21}$$

The three parts of the right-hand side of the equation are depicted in Fig. 2.5. By using a scalar product of equation (2.21) with $\boldsymbol{S}_i$ on both sides it is easy to show that the spin length is still conserved. However, as expected, the total energy of the system will not be conserved any more.

White and colored noise

As precised earlier, the limiting case of infinitely small correlation lengths is used. This formalism leads to a distribution which is independent of the frequency. Therefore, it is called white noise. Nevertheless, it also leads to unphysical phenomena because a distribution of white noise would lead to diverging power. A colored noise distribution $g(t)$ with a correlation length τ according to:

$$\langle g(t)g(t+t')\rangle \sim exp(-t'/\tau)$$

would be more reasonable. Such an approach would lead to a mathematically more complicated description of the problem. It can be shown [28] that for $\tau \to 0$ the noise $g(t)$ converges to the white noise. Therefore, for very short correlation times the white noise is a reasonable approximation.

2.4.2 Stochastic Differential Equation

In the previous section a thermal heat bath was introduced using a fluctuating term. The fluctuations were chosen to obey a white noise distribution for which every frequency contributes equally. Now the question arises how to mathematically treat differential equations with stochastic perturbations.

Stochastic Differential Equation

The general expression for a stochastic differential equation, such as equation (2.21) or equation (2.18) is typically written as:

$$\frac{dX(t)}{dt} = a(X(t),t) + b\left(X(t),t\right)\, f(t) \tag{2.22}$$

For simplicity, a one dimensional system is considered. The fluctuating term needs to fulfill the following properties to describe white noise :

$$\langle f(t)\rangle = 0 \tag{2.23}$$

$$\langle f(t)f(t')\rangle = \epsilon^2\delta(t-t') \tag{2.24}$$

The equation of motion (2.22) is often denoted as the Langevin equation with a deterministic drift term and a stochastic diffusive term. Integration of equation (2.22) over a small time interval dt leads to the standard stochastic differential equation with multiplicative noise used in stochastic calculus:

$$\begin{aligned} dX &= \int_t^{t+dt} a(X(t'),t')dt' + \int_t^{t+dt} b(X(t'),t')f(t')dt' \\ &= a(X(t),t)dt + b(X(t),t)dW(t), \end{aligned} \tag{2.25}$$

The term dW describes an infinitely small element of a Brownian motion which is defined as:

$$dW(t) = \int_t^{t+dt} f(t')dt' \tag{2.26}$$

W is, by mathematicians, often called Wiener process in honor of Norbert Wiener. Example paths of Wiener processes are given in Fig. 2.6. One can interpret equation (2.26) as a sum of Gaussian random variables. Thus, the mean value of $dW(t)$ will also be equal to zero:

$$\langle dW(t)\rangle = 0 \tag{2.27}$$

The variance of a Wiener process is given by:

$$\begin{aligned} \langle dW(t)dW(t)\rangle &= \int_t^{t+dt} dt_1 \int_t^{t+dt} dt_2 \langle f(t_1)f(t_2)\rangle & (2.28)\\ &= \int_t^{t+dt} dt_1 \int_t^{t+dt} dt_2\, \epsilon^2\, \delta(t_1 - t_2) & (2.29)\\ &= \epsilon^2 dt & (2.30) \end{aligned}$$

whose variance is proportional to ϵ^2 and linear in dt. If there is no overlap between the intervals $[t, t+dt]$ and $[t', t'+dt]$ (which is true for successive timesteps), one gets:

$$\langle dW(t)dW(t')\rangle = 0 \quad (2.31)$$

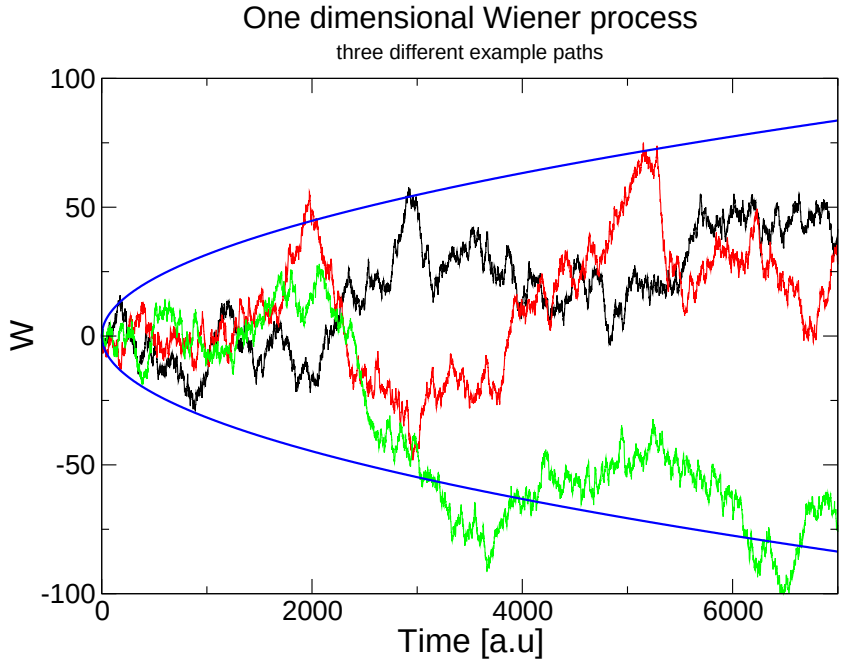


Figure 2.6: Three example paths (black,red, green) of a Wiener process with $\epsilon = 1$, the standard deviation (blue) is increasing with the square root of t.

Thus, there is no correlation in time between t and t'. The order of $dW(t)$ corresponds to the square root of the order of dt:

$$dW(t) = \epsilon f(t)\sqrt{dt} \quad (2.32)$$

This leads to crucial results for the integration of stochastic differential equations and is the origin of the Ito-Stratonovich controversy which will be discussed below.

Integration

The first term in equation (2.25) can be integrated straight-forwardly by using the Riemann integral. It is defined by a discretization of the time interval in N time steps t_i where the time step is given by Δt.

$$\sum_{j=1}^{N} a(\cdot, \delta\, t_j + (1-\delta) t_{j-1}) \cdot (t_j - t_{j-1})$$

It is well known that the Riemann integral converges for any choice of the value of the function $a(\cdot)$ between t_{j-1} and t_j in the limit of $\Delta t \to 0$. Therefore, δ can be chosen arbitrarily in the range $\delta \in [0, 1]$. In the limit $\Delta x \to 0$ the integral values converge to the same result.

A similar definition can be used for the stochastic part of equation (2.25) which is known as the Riemann-Stieltjes integral:

$$\sum_{j=1}^{N} b\Big(\delta X(t_j) - (1-\delta) X(t_{j-1}), t_j\Big) \cdot (W(t_j) - W(t_{j-1})) \tag{2.33}$$

As already mentioned, the choice of δ is arbitrary for an ordinary integral in the limit of $\Delta t \to 0$ and all results converge to the same result. However, this is not the case for the stochastic integral. It is even not guaranteed that such an integral converges. Although one can show that for the choice $\delta = 0$ and $\delta = 1/2$ the integral converges, it is not ensured that both choices lead to the same result. This problem is called the Ito-Stratonovich controversy [29, 30, 31].

The second part of equation (2.33) can be approximated for small values $\Delta t \to dt$ by:

$$b\Big(\delta\, X(t + dt) + (1-\delta)\, X(t), t\Big) dW(t)$$

The deviation in dt can be rewritten as a deviation in dX:

$$\begin{aligned} \delta\, X(t+dt) + (1-\delta)\, X(t) &= \delta\, (X(t) + dX(t)) + (1-\delta)\, X(t) \\ &= X(t) + \delta\, dX(t) \end{aligned}$$

Thus, the differential equation can be rewritten as follows:

$$dX(t) = a(X(t), t) dt + b\,(X(t) + \delta\, dX(t), t)\, dW(t) \tag{2.34}$$

Performing a Taylor expansion and summing up all contributions up to the second order

in $dW(t)$ one ends up with:

$$\begin{aligned} b(X(t)+\delta dX(t),t)dW(t) &= b(X(t),t)dW(t) + \frac{\partial b(X(t),t)}{\partial X(t)} dW(t)\,(\delta\, dX(t)) + ... \\ &= b(X(t),t)dW(t) + \delta\,\frac{\partial b(X(t),t)}{\partial X(t)} a(X(t),t)dt\, dW(t) \\ &+ \delta\,\frac{\partial b(X(t),t)}{\partial X(t)} b(X(t)+\delta X(t),t)(dW(t))^2 + ... \\ &= b(X(t),t)dW(t) + \delta\,\frac{\partial b(X(t),t)}{\partial X(t)} b(X(t),t)(dW(t))^2 \\ &+ \mathcal{O}(dW(t)^3) \end{aligned} \tag{2.35}$$

In the derivation, $dX(t)$ has once been substituted by a tailor expansion of equation (2.34) and dt has been substituted by equation (2.32). Inserting this term into equation (2.34) one gets:

$$dX(t) = a(X(t),t)dt + \delta\,\frac{\partial b(X(t),t)}{\partial X(t)} b(X(t),t)(dW(t))^2 \tag{2.36}$$

$$+b(X(t),t)dW(t) \tag{2.37}$$

$$= \left[a(X(t),t) + \delta \frac{\partial b(X(t),t)}{\partial X(t)} b(X(t),t)\epsilon^2\right] dt + b(X(t),t)dW(t) \tag{2.38}$$

$dW(t)^2$ was replaced by $\epsilon^2\, dt$. This can be justified by evaluating the quadratic mean of the difference between the increment dx of equation (2.36) and the increment $d\bar{x}$ of equation (2.38), which is given by:

$$\langle (dx - d\bar{x})^2 \rangle = \mathcal{O}(dt^2)$$

The difference of the quadratic mean decreases with an at least quadratic order towards zero, for $dt \to 0$ and can, thus, be neglected for an integration over a finite time.

Ito interpretation

Setting $\delta = 0$ results in the so-called Ito interpretation of the integral. It will be denoted as:

$$dX = a(X(t),t)dt + b(X(t),t)dW(t)$$

Stratonovich interpretation

Setting $\delta = 1/2$ yields to the so-called Stratonovich interpretation of the integral. To distinguish the two types of integral one uses the '$\circ$' symbol in the Stratonovich interpretation:

$$dX = a(X(t),t)dt + b(X(t),t) \circ dW(t)$$

The Ito and the Stratonovich interpretation of the integral can easily transformed into each other by introducing an additional term. The Stratonovich interpretation of the integral can be written in the Ito form as follows:

$$dX = \left[a(X,t) + \frac{1}{2}\frac{\partial b(X,t)}{\partial X} b(X,t)\epsilon^2\right] dt + b(X,t)dW(t)$$

Vice versa, the Ito interpretation of the integral can also be written in the Stratonovich form:

$$dX = \left[a(X,t) - \frac{1}{2}\frac{\partial b(X,t)}{\partial X} b(X,t)\epsilon^2\right] dt + b(X,t) \circ dW(t)$$

Depending on the interpretation, one needs to make a choice on the integral to use. Unfortunately, there is no general unique solution. In the literature dedicated to mathematical proofs and considerations, the Ito interpretation is often used because it fulfills important relations [32, 33], which are, for example, used in theoretical proofs. For practical purposes, it can be shown that the Stratonovich interpretation better describes dynamical spin systems [34].

Numerical Techniques

In the simulation code developed in this thesis, a weak Runge-Kutta (4th order) method for stochastic differential equations with small noise is used for solving the differential equation numerically. This method has been introduced by Milstein and Tretyakov [35]. Details are presented in the Appendix A.

2.5 Parallelization of the Code

Large scale simulations require significant computational resources. For studying the long time behavior for thousands of atoms it is useful to parallize the code to be able to make use of massively parallel supercomputers, which drastically speed up calculations. In the Forschungszentrum Jülich there are two supercomputer architectures available:

JUMP

The parallel supercomputer JUMP is an IBM p690 Cluster. It consists of 41 nodes with 32 processors each. Each node shares its memory of 128 GB among its processors. Therefore, such architectures are called shared memory systems. Every processor can basically access the memory of the whole node. However, accessing memory which belongs to a different node will suffer from big latency loss because the memory cannot be accessed directly. For simulations which use up to 32 processors JUMP can be used efficiently. For more then 32 processors each processor cannot access the shared memory directly. The information has to be sent over the network leading to an increase of computational time. A parallel code for shared memory systems is mostly written using OpenMP directives which can often be easily adapted to a code which is running on

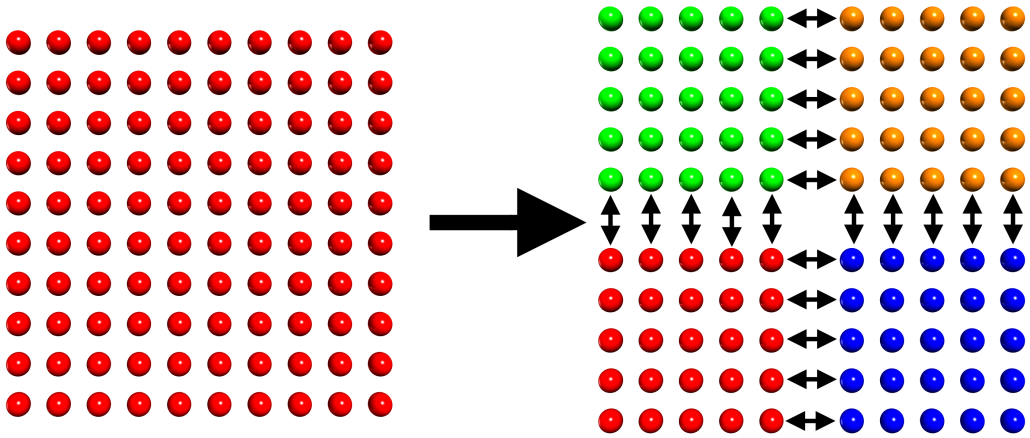


Figure 2.7: Distribution of work using domain decomposition, particles with the same color correspond to one processor, arrows between particles denote communication between processors.

a single CPU. The master thread will split the work to the N slave threads until the calculation of the loop has finished. After that the program will continue in a serial mode using the master thread. However, for a massively parallel code the use of JUGENE would be a better alternative.

JUGENE

JUGENE is a massively parallel supercomputer based on IBM's BlueGene and uses 65,536 PowerPC cores, which are clocked at 850 Mhz. Each CPU (which consists of 4 cores) has its own dedicated memory, which leads to a distributed memory system. Distribution and communication between each core has to be explicitly programed. This is done using the 'Message Passing Interface' (MPI).

Domain Decomposition

The parallelization of the spin-dynamics program has been performed using a domain decomposition (see Fig. 2.7), which works as follows:

- Information on simulation parameters, like temperature or particle positions is read by the master thread.
- The spin system is separated into as many domains as available cores. This is done in a way that the borders of each domain are minimized to reduce communication. The domains are assigned to the available cores and the particle positions as well as the spin positions are transmitted.

- Spins at the boundary of each domain which are called helos will need information on the neighboring domains. This item of information will be distributed to the demanding domain.
- Each thread calculates the change of the spin position in one Runge-Kutta step in its domain using the transmitted information of the neighboring domains.

A simple implementation is given by the following code:

```
--- Once  ----
    Do i=1,npp ! loop over all particles
      If (particle has neighbors outside domain )
        then ( Remember particle as a halo )
    END DO

--- Every Runge Kutta Time Step ---
    Do i=1,neighbors
      MPI_IRecv('Helos of Neighbor Domain i')
      MPI_ISend('Helos to Domain i')
      CALL RungeKutta()
    END DO
```

This scheme has currently been implemented for systems of spin chains. Future implementation will work for any kind of structures.

Scaling

In the ideal case a simulation run which is distributed among N threads should be N times faster compared to a calculation by a single thread. However, the practical simulation will never reach the theoretical estimations:

- Communication between processors costs time. The time for receiving information from another processor is orders of magnitude longer than a direct access to the local memory.
- Most parallel programs also contain serial parts. Therefore, just the parallel part can be distributed to each processor. The amount of serial code limits the amount of processors which can be efficiently used for a parallel run. This behavior refers to Amdahl's law which is explained in the Appendix A.3.

The efficiency of a parallel program is defined as:

$$E = \frac{t_1}{t_N \cdot N}$$

Here, t_1 denotes the simulation time on a single processor and t_N refers to the simulation time using N processors. A program with no performance loss would lead to an

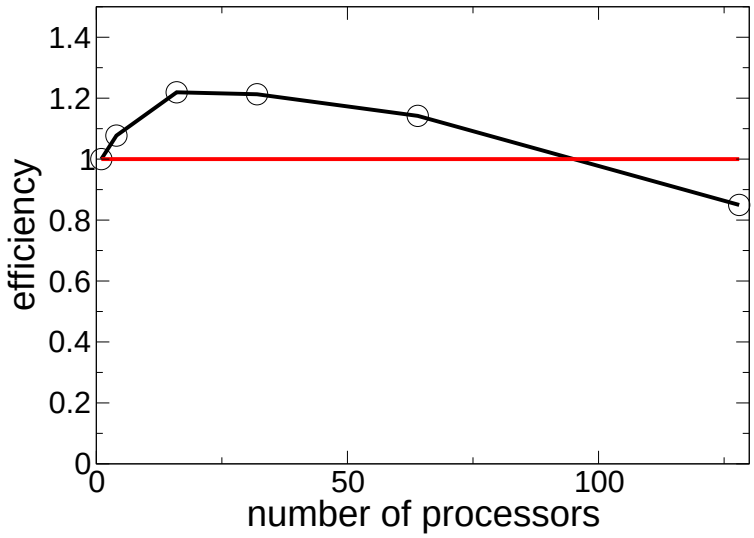


Figure 2.8: Efficiency of a parallel version of the spin-dynamics code running on JuGene at the Forschungszentrum Jülich for different number of processors.

efficiency of $E(N) = 1$. In Fig 2.8 the efficiency values for a chain with 131072 atoms is plotted versus the number of threads. For a moderate number of processors (less than 100), the efficiency exceeds the value for a perfect scaling($E(N) > 1$). This is due to cache-memory effects. The amount of spins each thread has to calculate will be reduced by increasing the number of processors. Therefore, the data does not need to be transferred to the cache as often which leads to a further decrease of computational time. It is observable in the figure that for parallel runs up to 128 processors there is almost no efficiency loss.

This is true for systems with a lot of atoms. If one decreases the number of atoms the amount of work of each processor will also decrease. Thus, communication is required more frequently which leads to an efficiency loss.

2.6 Testing the Algorithm

In this section a number of tests performed on the correctness and accuracy of the implemented spin-dynamics algorithm is described.

First of all the trajectory of spin systems has been followed to ensure that the precession of the spin is achieved as expected. Therefore, the precession trajectories of a single spin in a magnetic field for various temperatures is plotted. Fig. 2.3 (section 2.3.2) the trajectory of a single spin in a damped system without temperature was shown. It presents a smooth curve around the ground state which converges to the equilibrium

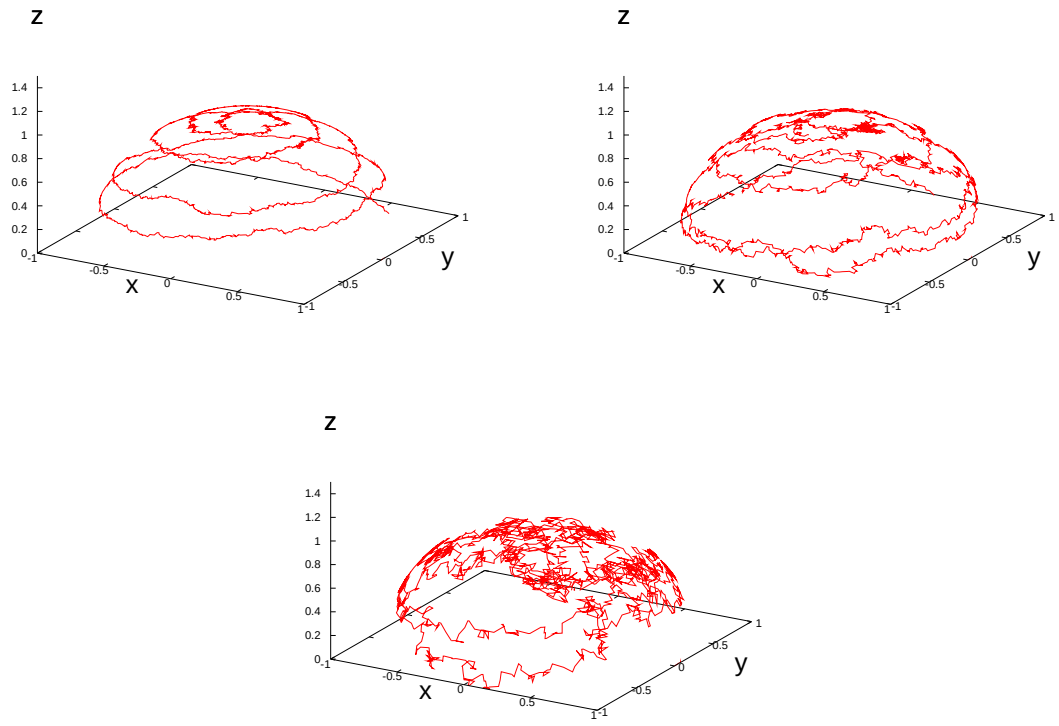


Figure 2.9: Trajectory for single spins in a magnetic field with damping and a temperature (starting from the upper left) of $T_1 = 0.02$, $T_2 = 0.1$, $T_3 = 0.5$.

position. Adding temperature to the model leads to a perturbation of the path. The spin will not converge to the ground state but move around it in a zone which depends on the temperature. On increasing the temperature this zone gets bigger and the perturbation due to the relaxation path is getting stronger. At one point, the trajectory followed by the spin will be independent, or uncorrelated in time, to the initial state. The correlation for a spin $\boldsymbol{S}_i$ in time can be described by the autocorrelation function:

$$C^0(t_0, t) = \langle \boldsymbol{S}_i(t_0) \cdot \boldsymbol{S}_i(t_0 + t) \rangle = \lim_{T \to \infty} \frac{1}{T} \int_{t_0}^{T} \boldsymbol{S}_i(t') \cdot \boldsymbol{S}_i(t' + t) dt'$$

where the brackets $\langle \cdot \rangle$ denote averaging over time. The integral in the autocorrelation function is approximated by a summation over a finite time. It converges exponentially to zero and the exponential factor is called the correlation time τ_c. One can show that the correlation time is inversely proportional to the temperature and inversely proportional to the damping parameter [36]. When averaging a physical quantity over time one should ensure that the time interval between subsequent samples is at least of the order of the correlation time.

A further test relies on the demand that, for an equilibrium state, the energy per spin should obey a Boltzmann distribution. This has been tested for various atomic

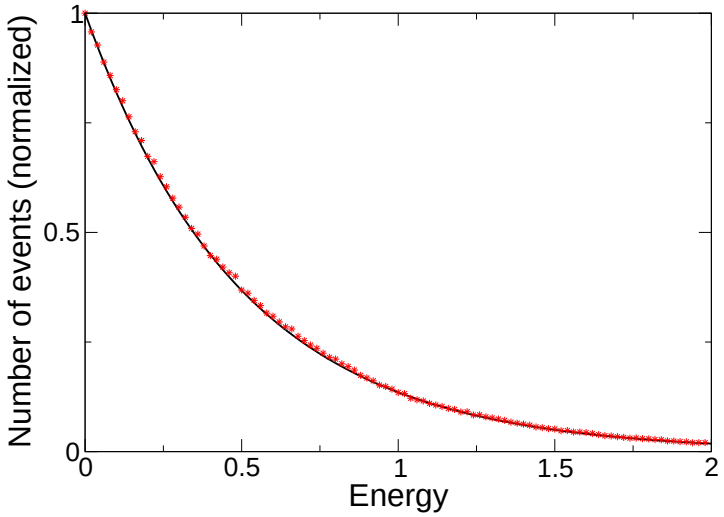


Figure 2.10: Histogram (red) of the energy distribution of a single spin in a magnetic field for an input temperature $T_{in} = 2g\mu_B H/k_B$ and a fit (black) of the numerical with an Boltzmann distribution which leads to a temperature of $T_{sim} = 1.988\, g\mu_B H/k_B$.

configurations. After letting the previous system converge to equilibrium, the energy values of the spin have been written out at each predefined time step. If enough statistical data is gathered and if the total collection time is sufficiently higher than the correlation length of the system the data can be plotted in a histogram and compared to a Boltzmann equation. Fig. 2.10 shows the distribution with an input parameter for the temperature of $T_{in} = 2g\mu_B H/k_B$. The fitting procedure calculated an exponential prefactor of $T_{sim} = 1.988\, g\mu_B H/k_B$. Hence, if one accounts for fitting errors due to finite histogram sizes and input data, the two values are in good agreement and one can say that the simulation code is able to describe the equilibrium states of spin systems.

The numerical errors in the model need to be kept small. After a simulation run it is, in general, hard to estimate the numerical errors. It depends mostly on the timestep used in the program, but it is also highly influenced by the simulation parameters (e.g. coupling constants, temperature, number of nearest neighbors). To estimate the numerical error in our simulations physical conservation laws are used. After every simulation run, the length of a single spin in the system is checked. Fig (2.11) shows the length of a spin in a cubic $10 \times 10 \times 10$ lattice with periodic boundary conditions over time. The length of the spin fluctuates due to numerical errors around the initial length. For high values of time steps, one can observe either a convergence of the total spin towards zero or a divergence to infinity. Otherwise, variations of the spin length can lead to systematical errors of the precession time of the spin. Thus, the length of each single spin is fixed to its initial value every at a time, which leads to a controllable numerical error.

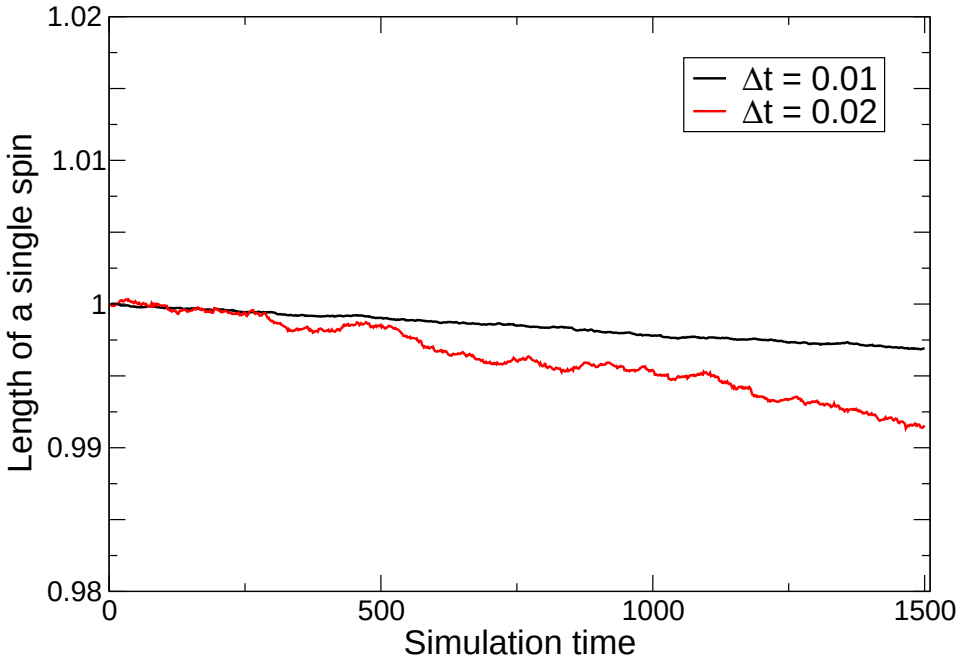


Figure 2.11: Length of a single spin during a simulation run of a $10 \times 10 \times 10$ cubic system. The deviation of the spin length from its initial length is an indicator for the accuracy of the numerical simulation.

3 Analytical Solutions

Beside performing atomistic spin-dynamic simulations we examine analytical solutions of the Landau-Lifschitz-Gilbert equation to get a better understanding of the physics behind it. The simulation of a temperature bath in a classical model leads to a system of coupled stochastic differential equations with multiplicative noise. An analytical solutions will lead to a Fokker-Planck equation which can describe the time evolution of a non-equilibrium probability distribution. However, we are first interested to find spin-wave solutions of the simplest form. Therefore, we used an undamped model with periodic boundary conditions and a nearest neighbor Heisenberg coupling. For this case, exact spin-wave solutions can be found but their superposition, unfortunately, does not solve the equation of motion. Therefore, as often done in physics, a further simplification is performed which relies on a linearization of the equation of motion. We were especially interested in the spin-mode solutions of a finite particle. Thus, a solver of the linearized equation of motion has been coded for an arbitrary finite clustered spin-mode system. In order to verify the correctness of the spin-dynamics simulation code, the obtained output has been compared to the analytical solutions and found to be in very good agreement.

3.1 Exact Spin-wave Solution

Although the Landau-Lifschitz equation is non linear spin-wave solutions exist. However, as mentioned previously, superpositions of two spin-wave solutions do not solve the equation of motion. Let us assume an equation of motion of the following form

$$\hbar \frac{\partial \boldsymbol{S}_i}{\partial t} = \frac{\partial \mathcal{H}}{\partial \boldsymbol{S}_i} \times \boldsymbol{S}_i$$

$$\mathcal{H} = \sum_i \sum_{j \in \mathrm{NN}} -\frac{1}{2} J \boldsymbol{S}_i \cdot \boldsymbol{S}_j - \sum_{i=1}^{N} K(\boldsymbol{e}_z \cdot \boldsymbol{S}_i)^2 - g\mu_B \boldsymbol{H}_{\mathrm{Zeeman}} \cdot \boldsymbol{S}_i \tag{3.1}$$

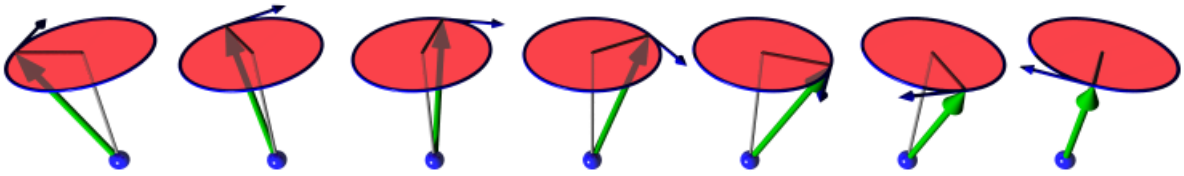


Figure 3.1: Spin-wave mode of an infinitely long chain.

where j runs over the nearest neighbors (NN) of spin i. To find spin-wave solutions [37], it is useful to use the following ansatz:

$$\boldsymbol{S} = S^z\boldsymbol{z} + A\boldsymbol{u}, \quad \boldsymbol{u} = Re(\begin{pmatrix}1\\ \mathrm{i}\\ 0\end{pmatrix} e^{\mathrm{i}(\boldsymbol{kr}-\omega t)}), \quad \boldsymbol{z} = \boldsymbol{e}_z$$

Here, the z-component of the spin is fixed to a constant value S^z and the radial part in the xy-plane with the amplitude A, described by the vector $\boldsymbol{u}$, is rotating around the z-component. A visualization of a spin-wave obeying this ansatz can be seen in Fig. 3.1. Using this notation, the effective field corresponding to the previous Hamiltonian, assuming an external field along the z-direction, can be written as:

$$H_{eff} = \frac{\partial \mathcal{H}}{\partial \boldsymbol{S}_i} = (\underbrace{-g\mu_B H^z_{\text{Zeeman}} - KS^z - nJS^z}_{H'})\boldsymbol{z} + (\underbrace{-JAD(\boldsymbol{k})}_{J'})\boldsymbol{u}$$

where n is the number of nearest neighbors. The function $D(\boldsymbol{k})$ is given by the sum of exponential functions of the nearest neighbor displacement vectors $\boldsymbol{\delta}$.

$$D(\boldsymbol{k}) = \sum_{\boldsymbol{\delta}\in \mathrm{NN}} e^{\mathrm{i}\boldsymbol{\delta}\cdot\boldsymbol{k}}$$

This sum leads to cosine functions. Thus, for a one dimensional chain the function is given by $D(k) = 2\cos(ka)$. The derivation of the terms for a simple cubic and a close-packed two dimensional structure can be seen in the appendix A.6.
It can easily be shown that the time derivative of the vector $\boldsymbol{u}$ is given by $\partial\boldsymbol{u}/\partial t = \omega \cdot \boldsymbol{v}$ where the introduced vector $\boldsymbol{v}$ is given by $\boldsymbol{v} = \boldsymbol{z} \times \boldsymbol{u}$. Inserting this into the equation of motion leads to the dispersion relation $\omega(\boldsymbol{k})$:

$$\begin{aligned}
\hbar\frac{\partial \boldsymbol{S}}{\partial t} &= (H'\boldsymbol{z} + J'\boldsymbol{u}) \times (S^z\boldsymbol{z} + A\boldsymbol{u}) \\
A\hbar\omega \cdot \boldsymbol{v} &= (S^zJ' - AH')\boldsymbol{v} \\
&= (Ag\mu_B H^z_{\text{Zeeman}} + AKS^z + AJS^z(n - D(\boldsymbol{k})))\boldsymbol{v} \\
\hbar\omega &= g\mu_B H^z_{\text{Zeeman}} + KS^z + JS^z(n - D(\boldsymbol{k}))
\end{aligned} \tag{3.2}$$

For an infinitely long chain the dispersion relation is shown in Fig. 3.2. It is easy to see that increasing the anisotropy constant or the external magnetic field leads to a shift of the dispersion relation to higher or lower frequencies. Increasing the length of the z-component of the spin or the coupling constant leads to a broadening of the dispersion.

It is interesting to note that beside the ferromagnetic ground state in which all spins are pointing in the z-direction, other stationary solutions can be found. For example, in a one dimensional system without a magnetic field, stationary state with $S^z = 0$ exists, where all spins are oriented in the xy-plane, with neighboring spins rotated by a constant angle. Nevertheless, this solution is unstable; small fluctuations will lead to a relaxation.
After a comparison to the analytical solutions, a perfect agreement was found with the trajectories provided by the simulation program.

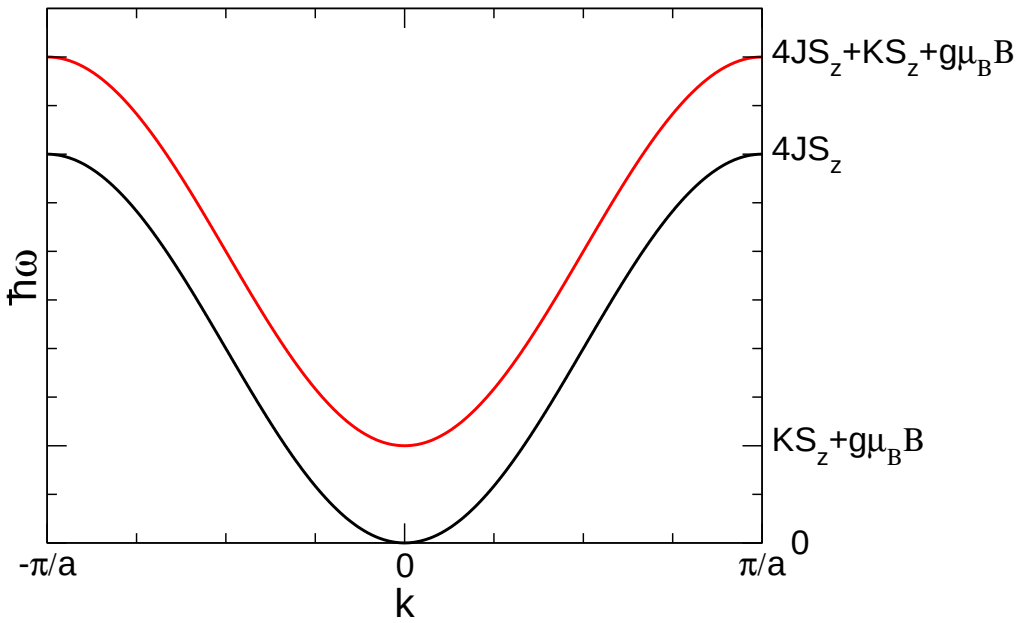


Figure 3.2: Dispersion of an infinite chain with (black) a ferromagnetic coupling and (red) including a magnetic field and an anisotropy.

3.2 Linearized Equation of Motion

We already found analytical solutions of the classical equation of motion, which are, however, restricted to periodic systems. To be able to solve the equation of motion analytically, we use an approximation for the limit of an almost ferromagnetic state. The equation of motion is linearized for perturbations around the ground state of a ferromagnetic Heisenberg system including an anisotropy term and a Zeeman term pointing towards the z-direction. The same procedure can be performed for an antiferromagnetic system which will lead to a different dispersion relation. Let us assume that for a ferromagnetic system the spins at the ground state are pointing towards the z-direction and no anisotropy or external fields are present. For small deviations around the ground state the z-component of the spin almost does not change. The effective magnetic field acting in spin $\boldsymbol{S}_i$ is given by

$$\boldsymbol{H}_{\text{eff}\,i} = \frac{\partial \mathcal{H}}{\partial \boldsymbol{S}_i} = \sum_{j=\text{NN}_i} -J_{ij}\boldsymbol{S}_j + g\mu_B H^z_{\text{Zeeman}}\,,$$

where the sum runs over the nearest neighbors (NN_i) of spin $\boldsymbol{S}_i$. The equation of motion for a spin system (2.13) can be transformed as follows:

$$\hbar\frac{\partial \boldsymbol{S}_i}{\partial t} = \frac{\partial \mathcal{H}}{\partial \boldsymbol{S}_i} \times \boldsymbol{S}_i = \sum_{j=\text{NN}_i} -J_{ij} \begin{pmatrix} S^y_j S^z_i - S^z_j S^y_i \\ S^z_j S^x_i - S^x_j S^z_i \\ S^x_j S^y_i - S^y_j S^x_i \end{pmatrix} + g\mu_B H^z_{\text{Zeeman}} \begin{pmatrix} -1 \\ 1 \\ 0 \end{pmatrix} \tag{3.3}$$

One can linearize the equation of motion for small derivations around the ferromagnetic ground state ($J_{ij} > 0$) in which all spins align collinear. In the limit where $S_x, S_y \ll |\boldsymbol{S}|$ one can assume that $S^z \approx |\boldsymbol{S}|$ and products of S_x and S_y vanish:

$$\hbar \frac{\partial \boldsymbol{S}_i}{\partial t} = \sum_{j=\mathrm{NN}_i} -J_{ij} S \begin{pmatrix} S_j^y - S_i^y \\ S_i^x - S_j^x \\ 0 \end{pmatrix} + g\mu_B H^z_{\mathrm{Zeeman}} \begin{pmatrix} -1 \\ 1 \\ 0 \end{pmatrix} \tag{3.4}$$

Linear Chain

For an infinite linear chain with spins at positions $x_i = a \cdot i$ the linearized equation of motion (3.4) can be simplified to:

$$\begin{aligned} \frac{\partial S_i^x}{\partial t} &= (JS/\hbar)(2S_i^y - S_{i-1}^y - S_{i+1}^y) \\ \frac{\partial S_i^y}{\partial t} &= -(JS/\hbar)(2S_i^x - S_{i-1}^x - S_{i+1}^x) \\ \frac{\partial S_i^z}{\partial t} &= 0 \end{aligned}$$

It can be analytically solved using the ansatz:

$$S^x = \mathrm{Re}\left(\exp(\mathrm{i}k \cdot x_i)\right), \quad S^y = \mathrm{Im}\left(\exp(\mathrm{i}k \cdot x_i)\right), \quad S^z = \mathrm{const} \tag{3.5}$$

The dispersion relation [19] has the same structure as the dispersion relation of the exact solution:

$$\hbar\omega = 2JS(1 - \cos(ka))$$

If one identifies the z-component of the spin with the magnitude of the total spin vector (which is done in the limit of the linearized model) then the two dispersions agree.

Finite Clusters

In the case of finite clusters with nearest-neighbor coupling the general form of the linearized equation of motion including a magnetic field H^z_{Zeeman} in z-direction can be determined by solving an eigenvalue problem. Therefore, for a finite system of N spins, it is useful to rewrite equation 3.4 in a matrix form :

$$\frac{\partial \boldsymbol{s}}{\partial t} = \mathcal{M} \cdot \boldsymbol{s}, \quad \boldsymbol{s} = (S_1^x, \ldots S_N^x, S_1^y, \ldots S_N^y)^T \tag{3.6}$$

Here $\mathcal{M}$ is a $2N \times 2N$ matrix, playing the role of a time evolution operator, and $\boldsymbol{s}$ is a $2N$-dimensional vector containing the x- and y- components of $\{\boldsymbol{S}\}$. The vector $\boldsymbol{s}$ is defined in a way that the resulting matrix $\mathcal{M}$ is a skew symmetric matrix ($(\mathcal{M})_{ij} = -(\mathcal{M})_{ji}$). This will prove convenient in further analysis. Equation (3.6) represents a system of coupled differential equations since the matrix $\mathcal{M}$ has off-diagonal elements. For a chain

of 5 atoms, which are coupled using periodic boundary conditions without external magnetic fields the matrix $\mathcal{M}$ is, for instance, given by :

$$\mathcal{M} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & -2 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & -2 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -2 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & -2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & -2 \\ 2 & -1 & 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -1 & 2 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 2 & -1 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & -1 & 2 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

It is possible to decouple the system of differential equations by decomposition of the matrix $\mathcal{M}$. Therefore, one needs to find a matrix $\mathcal{P}$ which satisfies $\mathcal{M} = \mathcal{P} \cdot \mathcal{D} \cdot \mathcal{P}^{-1}$ where $\mathcal{D}$ is a diagonal matrix. By using a coordinate transformation ($\boldsymbol{s} \rightarrow \boldsymbol{q}$) one can decouple the system of differential equations and each can then be simply solved:

$$\begin{aligned} \frac{\partial \boldsymbol{s}}{\partial t} &= (\mathcal{P} \cdot \mathcal{D} \cdot \mathcal{P}^{-1}) \cdot \boldsymbol{s} \\ \frac{\partial (\mathcal{P}^{-1} \cdot \boldsymbol{s})}{\partial t} &= \mathcal{D} \cdot (\mathcal{P}^{-1} \cdot \boldsymbol{s}), \quad \boldsymbol{q} = \mathcal{P}^{-1} \cdot \boldsymbol{s} \\ \frac{\partial \boldsymbol{q}}{\partial t} &= \mathcal{D} \cdot \boldsymbol{q} \end{aligned}$$

The matrix $\mathcal{P}$ is constructed by the eigenvectors $\boldsymbol{v}_i$ of $\mathcal{M}$ and the diagonal elements of $\mathcal{D}$ are given by its eigenvalues. As mentioned above, the matrix, $\mathcal{M}$ is a skew symmetric matrix, which then has only purely imaginary eigenvalues:

$$\mathcal{M}\boldsymbol{s} = \lambda \boldsymbol{s} \quad \Longrightarrow \quad \boldsymbol{s}^{\dagger}\mathcal{M}\boldsymbol{s} = \lambda |\boldsymbol{s}|^2$$

The absolute value of $\boldsymbol{s}$ is obviously real. By adjunction († =complex conjugate and transpose) we obtain:

$$\begin{aligned} \lambda^* |s|^2 &= (\boldsymbol{s}^{\dagger}\mathcal{M}\boldsymbol{s})^{\dagger} = \boldsymbol{s}^{\dagger}\mathcal{M}^{\dagger}(\boldsymbol{s}^{\dagger})^{\dagger} = -\boldsymbol{s}^{\dagger}\mathcal{M}\boldsymbol{s} = -\lambda |s|^2 \\ &\Longrightarrow \lambda^* = \lambda \end{aligned}$$

Therefore, the eigenvalue $\lambda = \mathrm{i}\omega$ has to be purely imaginary. The solution of the $2N$ differential equations is thus given by:

$$q_i(t) = a_i \cdot e^{\mathrm{i}\omega_i \cdot t}$$

where ω_i is the imaginary part of the diagonal element at position i of the matrix $\mathcal{D}$. The variables a_i denote complex numbers representing the initial condition of the system. A back-transformation of $\boldsymbol{q}$ leads to the complex solution $\boldsymbol{s} = \mathcal{P} \cdot \boldsymbol{q}$, which gives a superposition of the eigenstates:

$$\boldsymbol{s}(t) = \sum_i a_i \cdot \boldsymbol{v}_i \cdot e^{\mathrm{i}\omega_i \ t} \tag{3.7}$$

Equation (3.7) is the general solution of the differential equation (3.6). It is useful to further investigate solutions which belong to just one eigenfrequency, the spin-modes. We are interested in purely real solutions corresponding to spin-waves. Therefore, we need to find eigenmodes which can be combined to a real solution. Let us assume that a real matrix $\mathcal{A}$ has a complex eigenvector $\boldsymbol{v}$ corresponding to an eigenvalue w; then the complex conjugated $\boldsymbol{v^*}$,w^* are also eigenvector and eigenvalue of the matrix $\mathcal{A}$:

$$\mathcal{A} \cdot \boldsymbol{v} = w\boldsymbol{v} \Rightarrow (\mathcal{A} \cdot \boldsymbol{v})^* = (w\boldsymbol{v})^* \Leftrightarrow \mathcal{A} \cdot \boldsymbol{v^*} = w^*\boldsymbol{v^*}$$

Therefore, one can combine both to a purely real and a purely imaginary solution. These are linearly independent from each other and have the same eigenfrequency. One of the eigenvectors is simply a collective rotation of the spin vectors by 90 degrees around the $\boldsymbol{z}$-axis. Hence, a linear combination of both leads to a rotational invariance of the eigenvectors towards the ground state as one would physically expect.

Thus, one can leave out half of the eigenvalues since they simply represent a rotation of an already given solution. Finally, one ends up with N eigenvectors for a system of N particles.

Spin-modes of Clusters

To visualize the eigenmodes of a cluster we use a top-view aspect in such a way that the constant z-component of the spin is pointing out of plane towards the viewer. In other words the three dimensional space of the spin system is projected onto the xy-plane, which then shows the (small) deviations from the ground state can be visualized.

A first application is done on linear chains. Motivated by spin-polarized scanning tunneling microscopy experiments [38, 39] on triangular-shaped Co nano-islands on Cu(111) (Fig. 3.3), we also focus on the eigenmodes of triangular-shaped clusters. We, furthermore, investigate disc-shaped clusters on a (001) surface.

First of all, it is important to note that for a model without a magnetic field the eigenmodes do not depend on the strength of the coupling constant. As already discussed in the appendix A.2, a change in the coupling constant can be compensated by a rescaling of time which would then lead to a simple linear change of the eigen-frequency. In the following section, the eigenfrequency will be given in terms of energy $E = \hbar\omega$ in units of JS and for simplicity external magnetic fields are not considered.



Figure 3.3: Spin-resolved scanning tunneling microscope picture of Co nano-islands on Cu(111) [38].

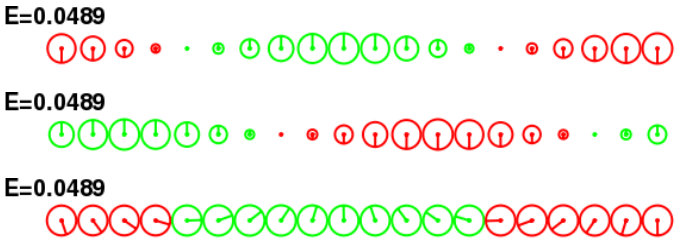


Figure 3.4: Two degenerated spin-modes of a chain with 20 atoms with periodic boundary conditions (first and second) which can be linearly combined to a spin-wave solution (bottom).

For testing purpose the solution of a linear infinite chain model have been compared to the analytical solution and found to lead to the same dispersion relation. Each eigenfrequency corresponds, for a periodic chain, to an at least two dimensional eigenspace. Hence a linear combination of two eigenmodes with the same eigenfrequency will still be an eigenmode. The linear combination can be done in a way that a traveling spin-wave solution with a constant amplitude in the xy-plane can be created. An example in which two eigenvectors are combined to a spin-wave is given in Fig. 3.4.

Here, however, we are interested in the behavior of finite systems. For finite chains all solutions that can be found are standing wave solutions. Each spin in an eigenmode has a xy-component of constant magnitude and precesses with its eigenfrequency mode around the z-axis. The finite length of the chains plays an important role since it determines the boundary conditions. Removing the periodic boundary conditions leads to a break of the translational symmetry and one is then not able to visualize the solutions in a dispersion picture. But the density-of-states is still a meaningful concept. In the limit of large chains a periodic and finite chain will have the same density-of-states. As discussed in the previous paragraph the degeneracy of eigenvectors can be used to

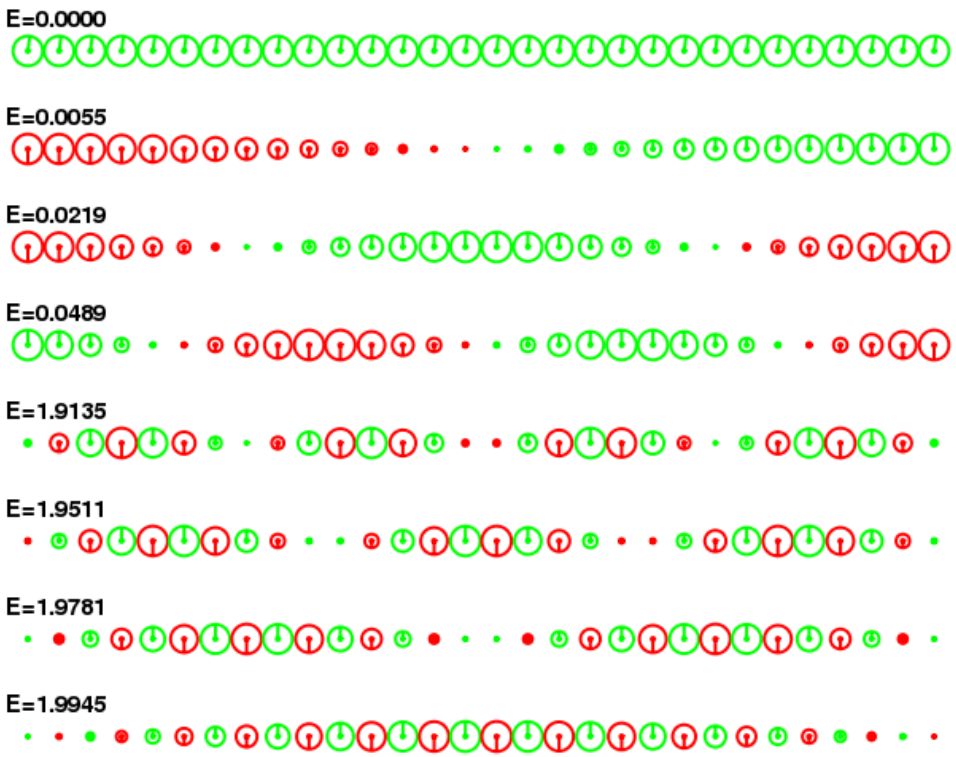


Figure 3.5: Projection of the spin positions into the xy-plane of one dimensional spin chains with open boundary conditions. The top four are the eigenmodes to the lowest the last four eigenmodes are the energetically highest eigenmodes.

construct spin-waves. However, for a finite chain, an energy gap between the degenerate eigenvectors is created. Therefore, they cannot be combined to a propagating solution anymore.

The density-of-states of a finite particle consists of sharp peaks in form of delta distributions. For a better visualization we used a Lorenzian broadening of the delta functions. The density-of-states plots have been calculated by using the following formula:

$$D(\omega) = \sum_{i=1}^{N} \frac{\Gamma}{\Gamma^2 + (\omega - \omega_i)^2}$$

The broadening factor Γ needs to be carefully chosen to get a meaningful figure. The eigenfrequencies, N in total, are given by the variables ω_i.

Two important aspects have to be considered while analyzing the eigenmodes (in the absence of a magnetic field). First of all, since the sum of all spins is conserved and we are investigating stationary solutions, the xy-components of the spins need to be zero for all eigenmodes with a non-zero eigenfrequency. Secondly, the normalized difference between the spins needs to be constant since all spins will have the same precession frequency in an eigenmode. Starting from equation (3.4), it can easily be shown that the precession frequency is proportional to the normalized difference between each spin and the average of its n neighboring spins which is defined as $\bar{\boldsymbol{S}}_i = \sum_{j \in \mathrm{NN}(i)} \boldsymbol{S}_j / n$:

$$\omega \propto \Delta S_i = \frac{|\boldsymbol{S}_i - \bar{\boldsymbol{S}}_i|}{|\boldsymbol{S}_i^{xy}|}$$

ΔS_i represents the difference of the spin S_i to the average of its neighboring spins normalized to the length of its xy-projection ($\boldsymbol{S}_i^{xy} = (S_i^x, S_i^y, 0)^T$). Thus, the lowest eigenmode with a normalized difference ΔS of zero corresponds to a vanishing eigenfrequency. The second lowest eigenfrequency can be determined in the following way. A boundary spin will be deviated from the ground state and the other spins will be changed in a way that the normalized difference is kept constant. This leads to a solution seen in Fig. 3.5. The resulting spin state will have a node in the center of the chain since the sum of all spins needs to be collinear to the z-axis. Other solutions with higher normalized differences can be formed by an increase of the number of nodes (which leads to an increase in the normalized difference). For high eigenvalues this leads to a "more anitferromagnetic" alignment of the spins (in the xy-plane) which are separated by nodes. This situation can be compared to the optical eigenmodes of harmonically coupled atoms in a chain in which the neighboring spins oscillate in antiphase. The highest mode is characterized by just two nodes at the boundaries of the chain. For the highest eigenvalue all neighboring spins are aligned antiparallel to each other in the xy-plane. It is important to note that in this case the deviation from the ground state in the xy-plane vanishes at the boundaries. This phenomenon is observed for all finite structures.

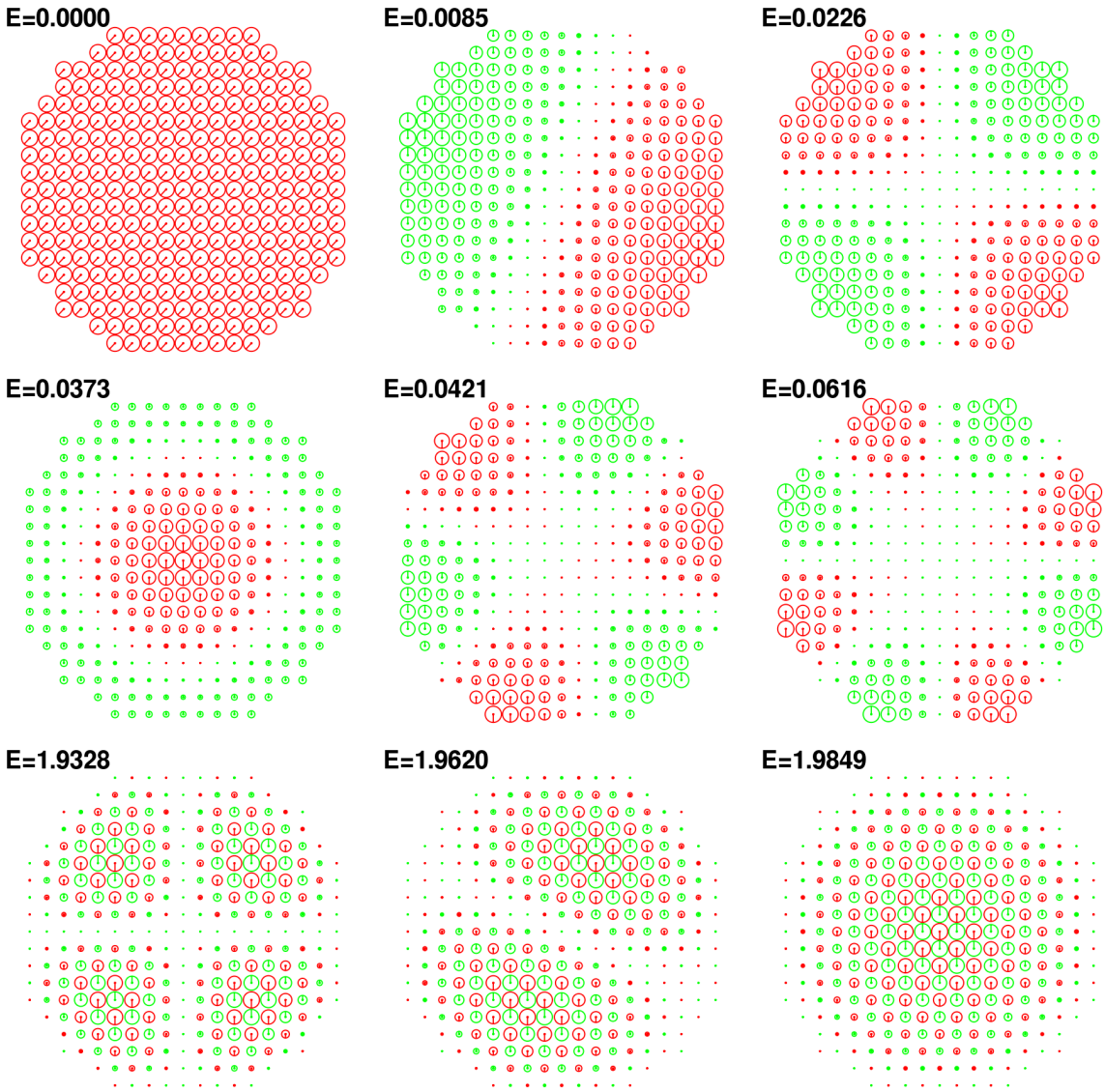


Figure 3.6: The xy-components of the spin eigenstates of a nano-disc. The first eigenmodes are the lowest and the last eigenmodes correspond to the highest eigenfrequency in the system.

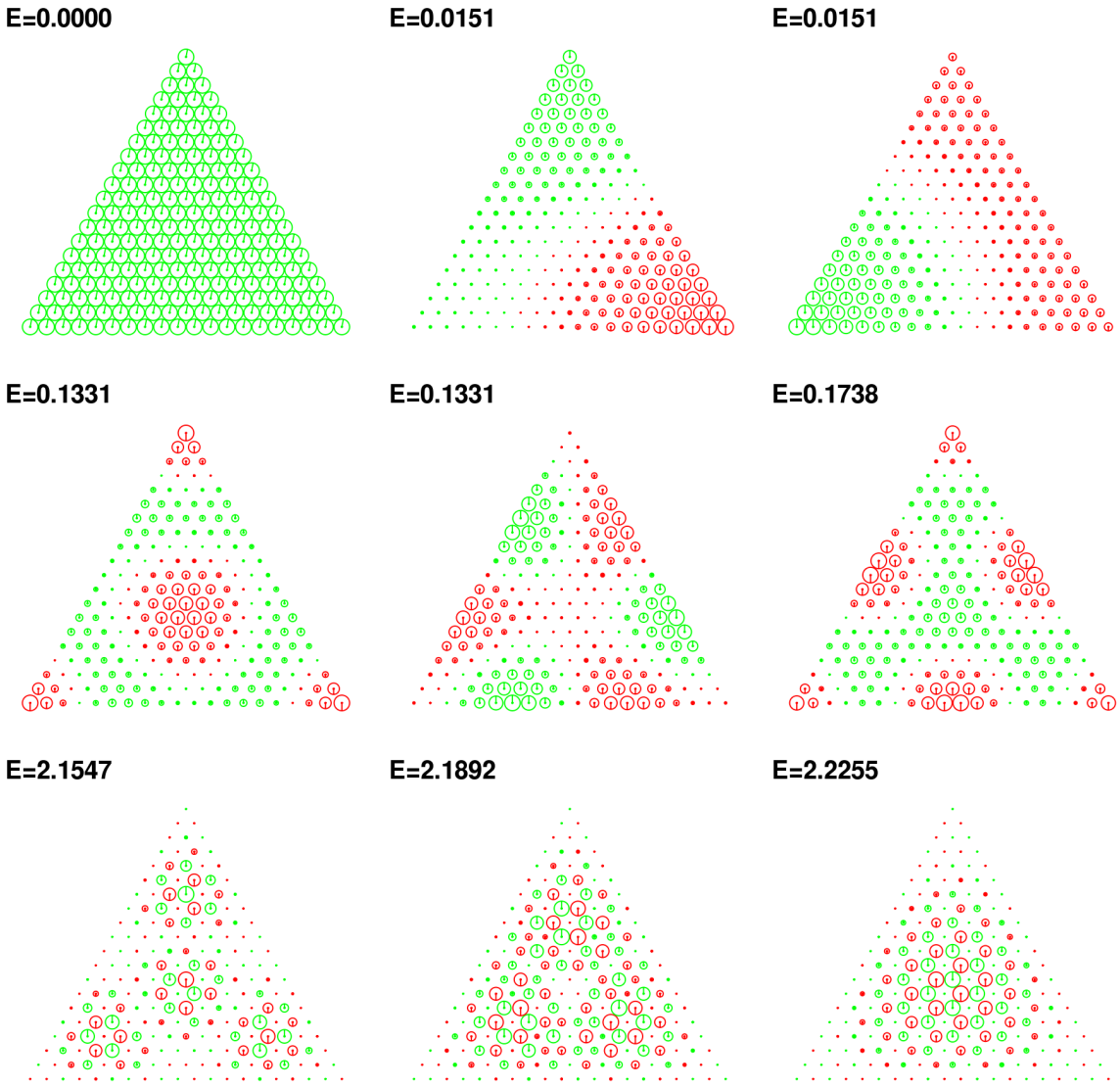


Figure 3.7: The xy-components of the spin eigenstates of a triangular two dimensional nanostructure. The first eigenmodes are the lowest and the last eigenmodes correspond to the highest eigenfrequency in the system.

Similar structures can be observed in two dimensional finite systems. Depending on the shape of the sample, nodes appear in different kind of directions. In Fig 3.6 a choice of the eigenmodes are given for a nano-disc. The first three are the lowest eigenmodes of the system whereas the last three are the eigenmodes corresponding to the highest eigenvalues. The formation of nodes for spin-modes of a two dimensional system can be understood by analyzing the node dimensional chain (seen in Fig. 3.5). Due to the quasi-rotational symmetry of the nano-disc one can observe nodes in the radial as well as in the angular part. For the lowest eigenfrequencies it is energetically favorable to first create nodes in the angular part(Fig. 3.6, $E = 0.0085$, $E = 0.0226$). For higher frequencies also structures with radial nodes appear(Fig. 3.6, $E = 0.0373$). At high frequencies optical eigenmodes appear, in which the xy-projection of neighboring spins is antiparallel. The number of nodes for the optical like modes again decrease with increasing eigenfrequency. One can observe a similarity between the three lowest and the three highest eigenmodes of the system. Both have the same structure of the node formation. The major difference is that the deviation of the spins from the ground state vanishes at the boundaries for high frequencies whereas the spins for low frequencies have their maximum at the boundary.

The triangular structured nanoisland (Fig. 3.7) has a rotational symmetry 120°. This is reflected in the structure of the eigenmodes. Eigenmodes belonging to a one dimensional eigenspace also have a rotational symmetry of 120°. Solutions whose dimension of the eigenspace is higher than one do not need to be rotationally symmetric. But one is always able to find combinations of eigenvectors in the eigenspace which correspond to a rotation by 120°.

A major difference between the nano-disc and the triangular structure is observable in the density-of-states plot (Fig 3.8). One can see that most of the eigenmodes for the nano-disc are located centrally between the lowest and the highest eigenmode whereas the peak of the triangular structure is shifted towards higher eigenfrequencies. This effect is, however, not related to the different shape of the sample, but to the lattice structure. Using equation (3.2) one can evaluate the dispersion relation for an infinite two dimensional close-packed and a simple tetragonal lattice structure and plot the density-of-states as shown in Fig. 3.9. One can clearly observe similarities especially for the location of the peak. For the cubic lattice structure the peak is located centrally, whereas the peak for the close-packed structure is shifted to the right as it was observed for the finite model in Fig. 3.8. This phenomenon can be explained using the Brillouin zone of the two structures. Indeed, the density-of-states can be determined by an integration over isolines which correspond to a constant energy:

$$D(E) \sim \int_{E(\boldsymbol{k})=E} \frac{d\boldsymbol{k}}{|\boldsymbol{\nabla} E(\boldsymbol{k})|}$$

Here $E(\boldsymbol{k})$ corresponds to the dispersion relation (in terms of energy). The approximate position of the peak can be estimated by using the following two assumptions:
First, it is assumed that the length of the isoline plays the dominant role and the gradient

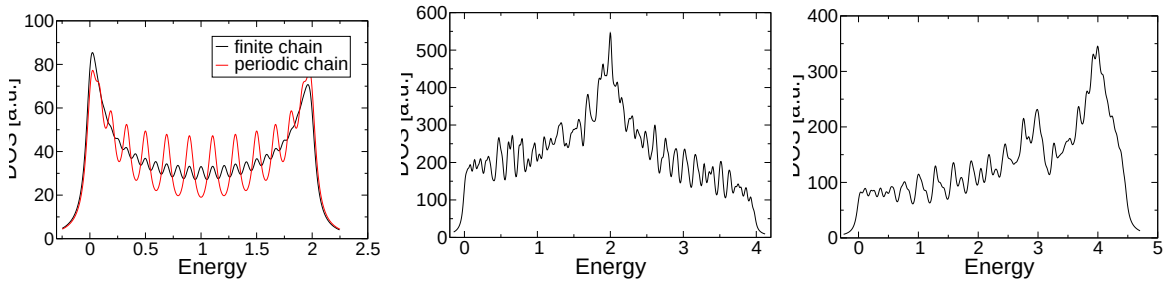


Figure 3.8: Density of states for a chain with 20 atoms (left), a nano-disc with a cubic structure and diameter of 20 atoms (middle), a close-packed triangular shaped nanoisland with a sidelength of 20 atoms.

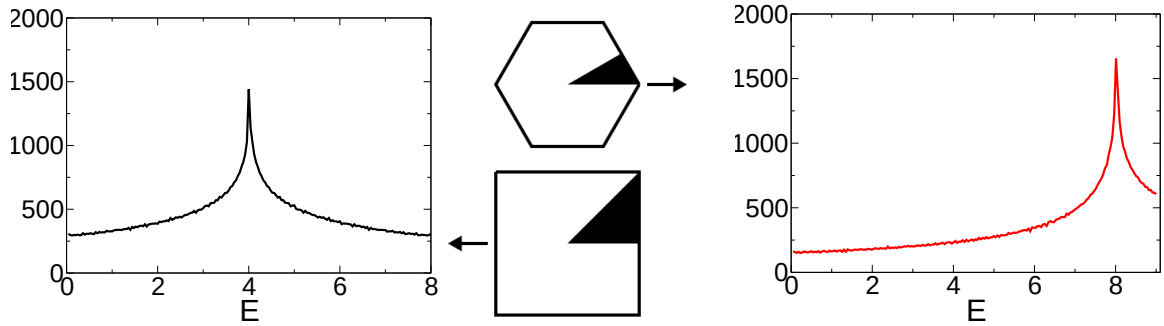


Figure 3.9: Density of states for a two dimensional infinite square lattice (left) and a two dimensional infinite lattice in a close-packed structure. The irreducible wedge of the Brillouin of both structures is plotted between both figures.

of the energy is not important. Thus, the number of states just depends on the length of the isoline. The second assumption is that the isoline can be approximated by a circle around the origin. By taking a close look at the Brillouin zones in Fig. 3.9 one can see that the longest isoline will, for the tetragonal lattice structure appear approximately half-way to the BZ boundary, centered between the highest and the lowest energy, while for the close-packed structure, the longest isoline will appear closer to the periphery of the BZ, i.e. closer to higher energies.

4 Multipole Methods

4.1 Introduction

While in small atomic clusters the exchange interaction plays the dominant role, the dipole-dipole interaction becomes more and more important with increasing number of spins. The dipole-dipole interaction serves to explain several physical properties such as shape anisotropies and the formation of domains. In this chapter we will focus on methods for the reduction of computational effort for the evaluation of the dipolar field.

The dipole-dipole interaction between two magnetic moments $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$ is

$$\mathcal{H}_{dip} = \boldsymbol{\mu}_2 \cdot \boldsymbol{B}(\boldsymbol{r}) \,,$$

where

$$\boldsymbol{B}(\boldsymbol{r}) = \frac{3\boldsymbol{r}(\boldsymbol{r} \cdot \boldsymbol{\mu}_1) - \boldsymbol{\mu}_1 \cdot r^2}{r^5} \tag{4.1}$$

is the magnetic field of a magnetic moment $\boldsymbol{\mu}_1$ in the origin evaluated at a position $\boldsymbol{r}$. The dipole-dipole interaction $\mathcal{H}_{\text{dip}}$ between the magnetic moments $\boldsymbol{\mu}_1$, $\boldsymbol{\mu}_2$ of two neighboring atoms is orders of magnitude smaller than the exchange interaction. However, it is a long-range interaction which decays proportional to $1/r^3$ where r is the distance between two atoms. Therefore, it is important to include the dipolar interaction in large systems.

Unless specially treated, the long-range behavior leads to a huge computational burden compared to the exchange interaction. In a nearest neighbor Heisenberg model the number of interactions per atom is independent of the total number of atoms, which leads to an almost linear increase of computational complexity with respect to the number of atoms. For a dipole-dipole coupling, due to its long-range behavior, all magnetic moments contribute to the local magnetic field acting on each spin. For a system of N atoms each spin is interacting with $N-1$ spins and, thus, a naive calculation would lead to a complexity proportional to $N(N-1)$. This quadratic increase of the computational time becomes crucial for large systems. However, as we shall see below, special treatments allow for considerable decrease of the computational effort with only a small and controlled expense in accuracy.

Quite a few problems in physics include a pairwise interaction between many particles. A well-known example is the evaluation of a $1/r$ potential like the gravitational N-body

problem which can be represented in the following way potential

$$\phi(\boldsymbol{r}_j) \sim m_j \sum_{i \neq j} \frac{1}{|\boldsymbol{r}_i - \boldsymbol{r}_j|} \cdot m_i \,,$$

where m_i, m_j denote the masses of particle i and particle j and $\boldsymbol{r}_i$, $\boldsymbol{r}_j$ represent their positions. This formula can be rewritten as a matrix-vector multiplication of the following form:

$$u_j = \sum_{i=1}^{N} K(x_i, y_j)\omega_i, \quad j = 1, ..M \tag{4.2}$$

For a coulomb potential, M and N will be the number of particles and ω_i is given by their mass. The matrix $(\mathcal{K})_{ij} = K(x_i, y_j)$ is called the kernel. A brute force calculation would lead to a complexity of $\mathcal{O}(N \cdot M)$. Such matrix-vector multiplications can be found in many different fields of physics like Coulomb interactions in condensed matter and plasma physics as well as in astronomy.

Example

The key issue is to find a way to separate the variables inside the kernel. The principle for reducing the complexity of the matrix vector product can easily be understood by the following example :

$$K(x_i, y_j) = (x_i + y_j)^2, \quad i = 1, ..N \quad j = 1, ..M$$

Brute force evaluation of the matrix vector product, equation (4.2), would lead to a complexity of $\mathcal{O}(N \cdot M)$. By using an algebraic transformation the product can be rewritten as follows:

$$\begin{aligned} u_j &= \sum_{i=1}^{N} (x_i + y_j)^2 \omega_i = \underbrace{\sum_{i=1}^{N} x_i^2 \omega_i}_{A} + y_j \underbrace{\sum_{i=1}^{N} 2x_i \omega_i}_{B} + y_j^2 \underbrace{\sum_{i=1}^{N} \omega_i}_{C} \\ &= A + y_j B + y_j^2 C \end{aligned} \tag{4.3}$$

The sums A, B and C do not depend on j. Hence, they just need to be evaluated once. This trick leads to a complexity of $\mathcal{O}(N + M)$. Such separated kernels are called degenerated kernels or kernels of finite rank:

$$K(x_i, y_j) = \sum_{k=1}^{p} \phi_k(x_i) \cdot \psi_k(y_j)$$

Once an expansion up to the order p is found the reduction of complexity is straightforward:

$$\begin{aligned} u_j &= \sum_{i=1}^{N} K(x_i, y_j)\omega_i = \sum_{i=1}^{N}(\sum_{k=1}^{p} \phi_k(x_i) \cdot \psi_k(y_j))\omega_i \\ &= \sum_{k=1}^{p} \psi_k(y_j) \underbrace{\sum_{i=1}^{N} \phi_k(x_i)\omega_i}_{A_k} = \sum_{k=1}^{p} \psi_k(y_j) A_k \end{aligned}$$

The moments A_k just need to be evaluated once and the amount of work is then reduced to $\mathcal{O}\left(p(N+M)\right)$. Although most of the kernels used in N-body problems are not degenerate, there are ways to find good approximations for the separation of the kernel. Two expansions which are used are the Taylor expansion and the expansion in spherical harmonics. However, these expansions are infinite sums which are commonly treated computationally by using cut-off parameters for the summation. This inaccuracy leads to an approximation of the matrix-vector product.

There are different methods which are based on finding approximations for evaluation of the matrix vector products (4.2). The most prominent ones are the Barnes-Hut (BH) tree algorithm and the Fast Multipole Method (FMM).

In 1968 Barnes and Hut [40] developed a so-called oct-tree algorithm in which the physical space is divided into 8 equally sized subcells. Each subcell is then again divided into 8 subcells until just one particle remains per cell. An expansion of the dipolar field is then formed for each cell at each level of refinement. The multipole moments of the different cells are calculated and then used to calculate the far field contribution of the magnetic field. The complexity of the problem can be reduced by this algorithm from $\mathcal{O}(N^2)$ to order $\mathcal{O}(N \log N)$.

Greengard and Rohklin [41, 42, 43] later on developed independently an algorithm similar to the BH tree-code which has some further refinements and is called the Fast Multipole Method. They used high order cluster-cluster interactions to transfer multipole moments for the far field as well as for the near field between separated regions of space. Greengard and Rohklin could show that the complexity can be reduced to $\mathcal{O}(N)$. However, the prefactor due to the high order expansions is much larger than the prefactor of the BH algorithm.

4.2 Barnes-Hut Algorithm

The algorithm used in this work is based on the Barnes and Hut algorithm combined with some elements of the Fast Multipole Method (FMM). There is no general rule which

algorithm to use. The choice depends on the application. For high precision calculations the FMM is the preferred method. But for applications where low precision calculations are sufficient and the particle distribution may be random the BH tree-code is competitive.

The basic idea of the Barnes and Hut tree-algorithm is to use coefficients of a far field expansion, the multipole moments, instead of evaluating the dipolar field arising from every single magnetic moment. The whole simulation region is subdivided into a fine grid of cells, to a level of refinement which is needed for an accurate calculation of the interaction between nearby cells. Nevertheless, for far away regions it is sufficient to use larger cells to obtain accurate results. Therefore, parent cells are formed by successively joining 2^d smaller neighboring subcells. Here, d denotes the dimension of the system. This way, an hierarchical cell structure with different levels is formed which is used to evaluate the dipolar field. The cells of different size correspond to different hierarchy levels, but cells within a level are all equally sized. The multipole moments of the parent cells can be evaluated by joining the multipole moments of its subcells. The magnetic field acting on each spin can then be determined by making use of different sized cells. There are different ways, based on 'multipole acceptance criteria' (MAC), on deciding the size of each cells which is used for the evaluation of the magnetic field
We proceed by explaining the algorithm used in our multipole code. It basically consists of the following steps:

Tree Build

First of all, a multilevel refinement of the simulation region is formed by subsequently dividing the d-dimensional space into 2^d subcells. Fig. 4.1 shows the separation of a two-dimensional simulation region in three levels. A subsequent separation of the simulation cell into four equally sized subcells can be seen in the figure. The separation of space can be visualized by a tree structure in which every node corresponds to a simulation cell. Its four subcells are called child nodes (see Fig. 4.1 middle) whereas the nodes from which child nodes are derived are called parent nodes. In fact, the tree structure is also used as a data structure in the multipole code. Each element in the data tree structure and the term 'node' will be interchangeably used for an element of the tree data structure and the visualization of the cell refinement. The case of a three dimensional space works in an analogous way, by refining each cell into eight subcells corresponding to an oct-tree structure. The tree starts with a root cell which represents the whole simulation region. It is refined into eight child nodes which contain information[1] on the subcells. The sub-tree structure will then be further refined if a child cell contains particles. Hence, a partly filled oct-tree is created which leads to a flexible algorithm for the calculation of one-, two- or three-dimensional structures. If the number of particles per subcell is below a predefined value the refinement of the sub-tree stops. Nodes which

[1] like for instance physical information such as coordinates of the particle center or multipole moment coefficients and technical data like keys of each node

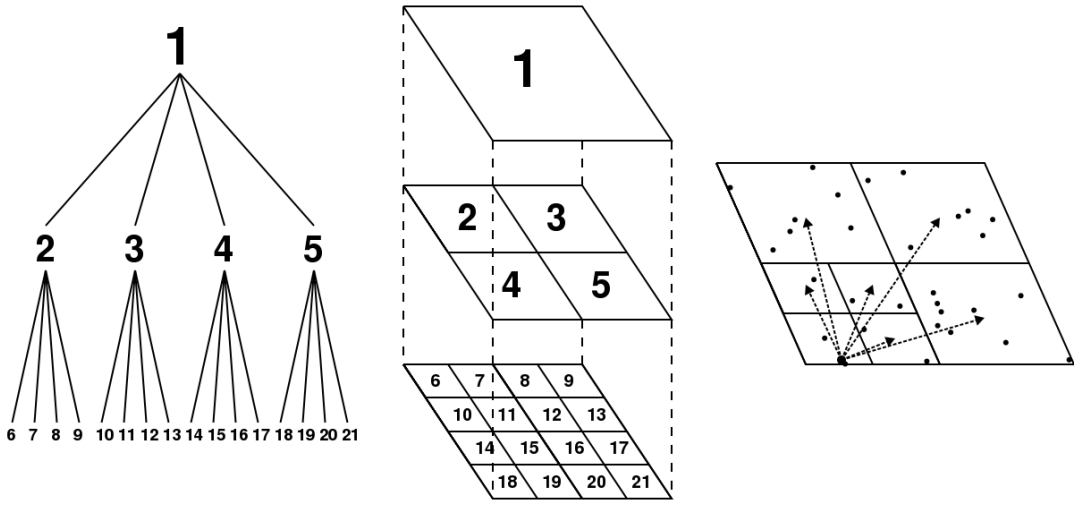


Figure 4.1: A tree structure (left) corresponding to a space separation (middle) of a two dimensional system into four subcells. The multipole moments of the cells are then used to evaluate the dipolar field of a particle system (right).

do not contain any child nodes are called leave nodes.

There are different ways to store a tree structure in memory. One can think about a tree data type in which every entry contains a pointer which states to the memory address of its children. This data type would lead to a very efficient use of the memory available. However, the access time will scale with $\mathcal{O}(\log N)$ since for every access of an element the code needs to walk through the pointer tree structure. The code will also be difficult to parallelize. Therefore, a powerful numbering scheme with excellent properties for a future parallelization is used, as explained in section 4.3.

Multipole Expansion

The multipole moments of the cells are first calculated at the leaves. Then the multipole moments of their parents are determined by joining multipole moments of the leave nodes. This procedure can successively be performed up the root and will be explained in detail in section 4.4. This algorithm leads to an efficient evaluation of the multipole moments.

Different basis functions can be used to expand the magnetic field. The most commonly used expansions are the Cartesian Taylor expansion and an expansion in spherical harmonics. The expansion in spherical harmonics is analytically known and are, therefore, often used in FMM codes. It is often argued that an expansion in spherical harmonics has less basis functions to a given expansion order l. This is true for highly accurate calculations ($l = 20$). We, however, use low-order ($l < 5$) expansions for which the number of basis functions is at the same order of magnitude (see section 4.4) as

the expansion in spherical harmonics. The basis functions are purely polynomial which leads to rapid evaluations of the fields. Other simulations [44] show that a Cartesian expansion can be faster comparing the simulation wall clock time at a constant accuracy level. Thus, an expansion in Cartesian coordinates has been chosen. Details on the expansion method which is used are presented in section 4.4.

Interaction List

Once the tree is built up and the multipole moments of each cell in the tree structure are calculated one needs to determine which cells and cell sizes to use for the evaluation of the magnetic field. This is done by creating a so-called interaction list for each atom containing the labels of all nodes which will be used to determine the magnetic field. As already mentioned before it will be sufficient to use the expansion of a low level node (which corresponds to a large region) for regions which are far away from the evaluation point. For nearby regions one needs to use small cells which correspond to high level nodes to keep the required accuracy.

The choice on which node to include in the interaction list is based on a 'multipole acceptance criterion' (MAC). The interaction list for a given point is constructed in the following way: Starting from the root each node will be tested for fulfilling the MAC. If the MAC is fulfilled then the node will be included into the interaction list. If not, then the refinement of the node will be used and the child nodes will be tested for fulfilling the MAC. If a leave node is reached and the MAC is not yet fulfilled, then the leave node will be included into a direct evaluation list in which, later on, the magnetic field will be calculated for each particle.

There are different formulations of the multipole acceptance criterion. In this work we use the original criterion of Barnes and Hut, which reads

$$\epsilon > \theta = s/d$$

where ϵ denotes the accuracy parameter and θ the opening angle which is defined as the fraction of the size of the cell s and the distance d to the evaluation point. If the fraction is smaller than ϵ the node is accepted, otherwise it is rejected. An interaction list for different values of the optimal opening angle can be seen in Fig. 4.2. One needs to find a good trade-off between accuracy and simulation time. A small opening angle leads to higher accuracy but also higher computational time. Reasonable values which are commonly used are around $\epsilon = 0.5$.

Fig. 4.3 shows the interaction list of the bold particle in the upper left corner of a 40×40 two dimensional system. Red boxes correspond to a direct evaluation while for blue boxes multipole expansion is used for evaluating the magnetic field. Similarly, in Fig. 4.4 the interaction list of a particle (red) is plotted for the case of a three dimensional space. The red and the green boxes correspond to a direct evaluation, while the multipole moments are used in the gray boxes.

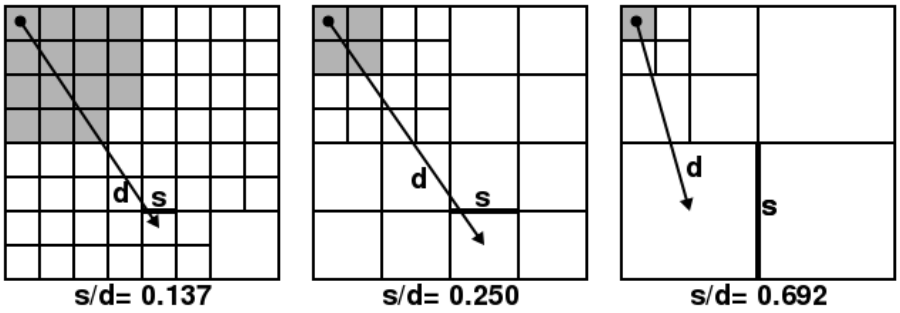


Figure 4.2: Visualization of three interaction lists of a particle comparing three different opening angles. White boxes fulfill the MAC, gray boxes do not.

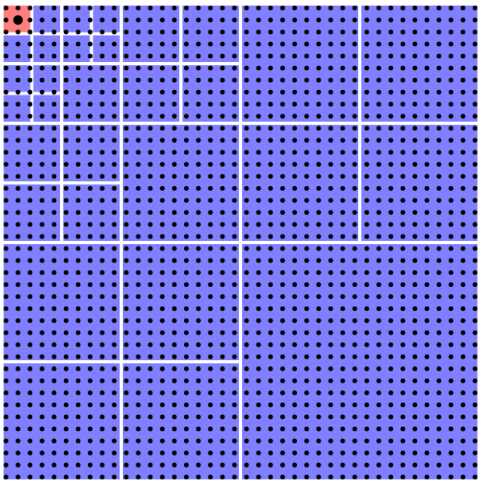


Figure 4.3: Visualization of the interaction list of a $40 \times 40 \times 1$ system for the bold atom. Blue boxes indicate regions which use a multipole expansion, red indicates regions which will use direct calculation for the magnetic field (direct simulation output picture).

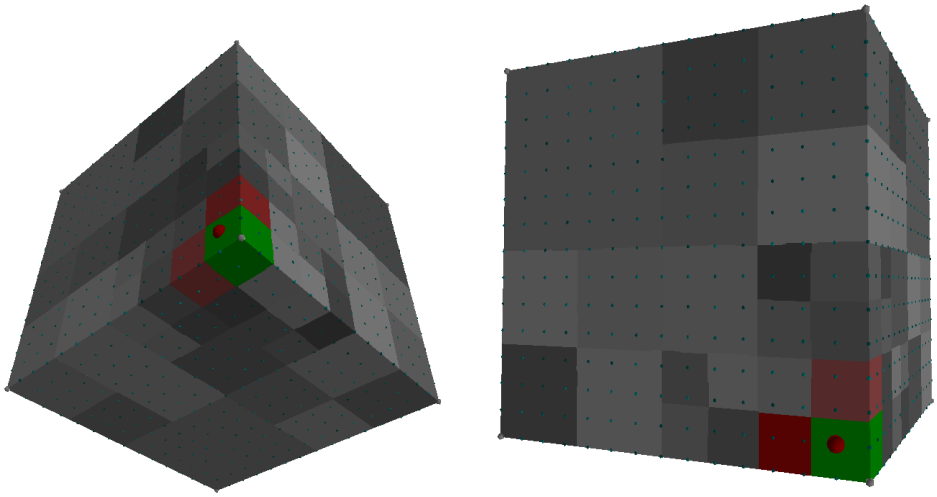


Figure 4.4: Representation of the interaction list for the red particle. Red and green boxes correspond to direct evaluation of the dipolar field of the particles. Gray boxes use the dipolar moments for the evaluation of the field.

Force Summation

The magnetic field acting on each particle is calculated. First of all, a direct calculation of the magnetic field by particles of cells in the direct evaluation list is performed. Then the magnetic field of all cells in the multipole interaction list is calculated using their multipole moments. This is the most time consuming part of the program.

4.3 Key Mapping

The aim of this section is to present a suitable scheme to label each node in the tree structure. An indexing scheme is used which maps the three dimensional space to a one dimensional array (or a one dimensional path) for every level of refinement. Every particle in the system is labeled with a (not necessarily unique) key which is computed using its coordinates.

The indexing scheme is easy to understand in a one dimensional example in which particles at the position x_i are distributed in the range $[0, 1]$. Each node of the tree structure is labeled using binary numbers, starting with the root key which is labeled with '1'. The keys of the two children nodes are then determined as follows: First the root key is shifted by one bit to the left. Secondly a '0' is added for the left child and a '1' for the right child. Thus, the left child will be labeled with '10' and the right child with '11'. This procedure can be seen in Fig. 4.5 and can successively be applied up to an arbitrary level. This algorithm can also be used to determine the parent of a node by reversing the algorithm by performing a bitshift to the right.

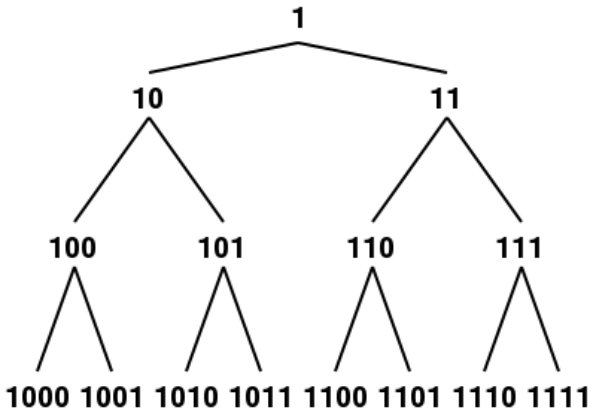


Figure 4.5: Indexing of a one dimensional tree refinement using binary numbers.

The advantage of this labeling structure is that one does not need to perform this algorithm to obtain the nodes which a particle belongs to. One can easily determine the nodes at each level l which contains the particle at position x_i by using the following equation:

$$key = \text{int}(x_i \cdot 2^l) + 2^{l+1} \tag{4.4}$$

The function int$(\cdot)$ returns the integer value of the variable given. A particle at position $x = 0.4$ will, for example, according to the previous equation, be labeled with '10' at level $l = 1$ and with '101' at level $l = 2$. Furthermore, one can determine the parent node of node '101' by using a bitshift to the right and taking the integer value ('101'→'10').

This procedure can be generalized to a three dimensional distribution of particles in a unit cube. The key of each node can be created by joining the keys of each spacial direction calculated using equation (4.4) (excluding the leading '1') which can be seen in Fig. 4.6. The key will then be stored in a 64-bit integer. Thus, the maximum number of integers for each spacial direction is given by $l_{max} = 21$ which is the maximum level which can be used by this formalism. The key joining can mathematically be described by the following equation:

$$\text{key} = 8^{l_{\max}} + \sum_{j=0}^{l_{\max-1}} 8^j (4 \cdot \text{bit}(z \cdot 2^j, j) + 2 \cdot \text{bit}(y \cdot 2^j, j) + 1 \cdot \text{bit}(x \cdot 2^j, j))$$

The function bit(k, j) simply pics out the jth bit of k starting with $j = 0$. The key for the maximum number of levels will be called particle key and will first be calculated and assigned to the particle in the code.

Sorting the keys leads to a path for the particles in the simulation region, which can be seen in Fig. 4.7. The path leads to an optimal separation of the three dimensional space into most compact regions which is called Morton ordering or Z-ordering. The

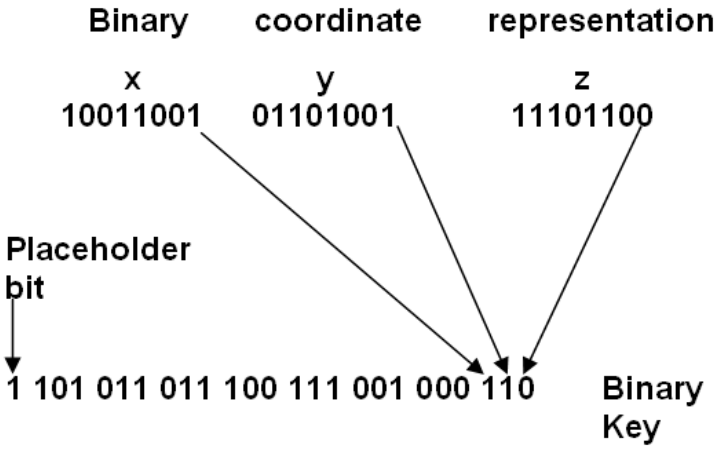


Figure 4.6: Procedure for the generation of the keys assigned to each particle.

Morton path is important for a future parallelization of the multipole code.

As discussed before, by knowing a key of a particle one already has the information of all its ancestor nodes. The keys of parent nodes of each particle are for an oct-tree simply created by performing three bit shift to the right instead of one. This procedure can again be used until the root node which is label with 1 is reached.

$$parentkey = \text{rightshift}(\ key,3)$$

The same procedure can be used to find child nodes of a particle. One just needs to perform a left bitshift by three and then add a three bit integer number $(0-7)$:

$$childkey = \text{leftshift}(\ key,3) + \ child, \qquad child = 0,..7$$

Up to a certain level the keys can also be used as addresses for storing information on the cells into a one dimensional array. This procedure leads to an access time of $\mathcal{O}(1)$ while a tree data structure would lead to an access time of $\mathcal{O}(\log N)$.

However, for big systems or just partly filled trees one would waste a lot of memory by using the particle key as the origin in the storage array, especially for one or two dimensional systems. A concept known as hash function can be used to save storing memory. Further information can be found in [45]. If one knows the particle positions of the system in advance special optimized hash functions can be employed.

4.4 Expansion of the Dipolar Field

The kernel of the matrix-vector multiplication can be expanded in different basis functions. Two expansions are commonly used. One is an expansion in spherical harmonics

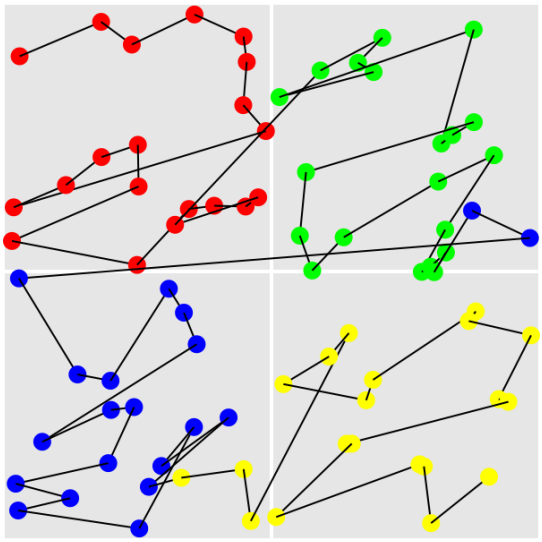


Figure 4.7: Key mapping path through a system of 80 random distributed particles. The colors indicate different domains which are calculated by different processors in parallel.

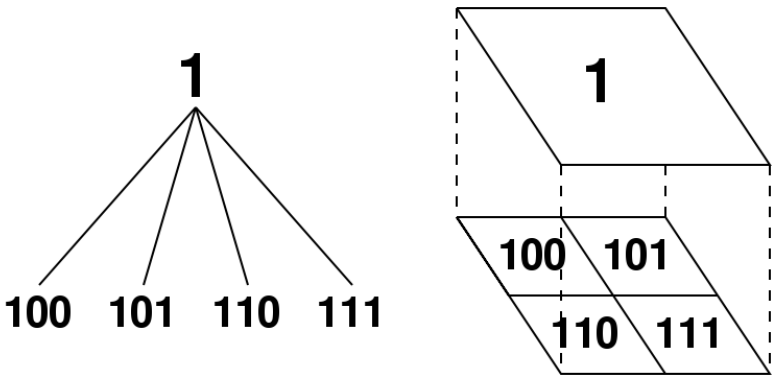


Figure 4.8: Scheme used for labeling each simulation region with unique keys. Parent and child keys can easily be found by performing two bitshift operations.

Y_{lm} which is often used in Fast Multipole Methods and has the advantage that the expansion functions are analytically known [46]:

$$\frac{1}{|\boldsymbol{r}-\boldsymbol{r}'|} = \sum_{l=0}^{\infty}\sum_{m=-l}^{l} \frac{4\pi}{2l+1}\frac{r'^l}{r^{l+1}} Y_{lm}^*(\theta',\phi')Y_{lm}(\theta,\phi)$$

Here, $Y_{lm}(\theta,\phi)$ and $Y_{lm}^*(\theta',\phi')$ express the spherical harmonics as a function of the spherical coordinates $\boldsymbol{r}(r,\theta,\phi)$ and $\boldsymbol{r'}(r',\theta',\phi')$ for the case $r > r'$. The Taylor expansion in comparison is given by :

$$\frac{1}{|\boldsymbol{r}-\boldsymbol{r}'|} = \sum_{l=0}^{\infty} \sum_{\substack{l_x,l_y,l_z=0 \\ l_x+l_y+l_z=l}}^{\infty} \frac{1}{l_x!}\frac{1}{l_y!}\frac{1}{l_z!} \left((-1)^{l_x+l_y+l_z} \frac{\partial^{l_x}}{\partial x^{l_x}} \frac{\partial^{l_y}}{\partial y^{l_y}} \frac{\partial^{l_z}}{\partial z^{l_z}} \frac{1}{|\boldsymbol{r}|} \right) x'^{l_x} y'^{l_y} z'^{l_z}$$

It is often argued that it is more sufficient to use an expansion in spherical harmonics for high values for the cut-off parameter ($l = 20$) because less basis function have to be used compared to a Cartesian expansion. Both expansions should lead to an almost equal accuracy for a given value of l. However, for spin-dynamics simulations an evaluation of the magnetic field with a low precision ($l < 5$) is sufficient. Therefore, a Cartesian Taylor expansion is used. In the low precision regime the increase of the number of basis functions [47] of the Cartesian expansion compared to the spherical harmonics expansion is fairly small, as we can see in the following table:

order of l	0	1	2	3	4	5	6	7	8
number of spherical harmonics	1	3	5	7	9	11	13	15	17
number of Cartesian expansion functions	1	3	6	10	15	21	28	36	45

The advantage of the Cartesian expansion is that the basis functions are purely polynomial which leads to a rapid computational evaluation of the magnetic field.

The magnetic field at a point $\boldsymbol{r}$ induced by a point dipole $\boldsymbol{\mu}$ at the origin is given by equation (4.1). Our aim is now to separate the variables using a Taylor expansion. It is useful to rewrite the magnetic field in terms of $1/r$ to make use of the well-known expansion of a Coulomb potential.

Equation (4.1) can be written in terms of the vector potential $\boldsymbol{A}$:

$$\boldsymbol{A}(\boldsymbol{r}) = \frac{\boldsymbol{\mu}\times\boldsymbol{r}}{|\boldsymbol{r}|^3}, \quad \boldsymbol{B} = \boldsymbol{\nabla}\times\boldsymbol{A}$$

The vector potential can be transformed such that it depends on an $1/r$ potential:

$$\boldsymbol{A}(\boldsymbol{r}) = \boldsymbol{\mu}\times\frac{\boldsymbol{r}}{|\boldsymbol{r}|^3} = -\boldsymbol{\mu}\times\boldsymbol{\nabla}\cdot\frac{1}{|\boldsymbol{r}|} = \boldsymbol{\nabla}\times\boldsymbol{\mu}\cdot\frac{1}{|\boldsymbol{r}|} = \boldsymbol{\nabla}\times\boldsymbol{\phi} \tag{4.5}$$

$$\boldsymbol{\phi} = \frac{\boldsymbol{\mu}}{|\boldsymbol{r}|} \tag{4.6}$$

The vector field $\boldsymbol{\phi}$ defines a magnetic potential. It has a similar form as the Coulomb potential with the difference that the electric charge is a scalar and the magnetic 'charge' is a three dimensional vector.

One can easily derive the magnetic field to:

$$\begin{aligned} \boldsymbol{B} &= \boldsymbol{\nabla} \times (\boldsymbol{\nabla} \times \frac{\boldsymbol{\mu}}{|\boldsymbol{r}|}) \\ &= \boldsymbol{\nabla}(\boldsymbol{\nabla} \frac{\boldsymbol{\mu}}{|\boldsymbol{r}|}) - \boldsymbol{\nabla}^2 \frac{\boldsymbol{\mu}}{|\boldsymbol{r}|} \\ &= \boldsymbol{\nabla}(\boldsymbol{\nabla} \frac{\boldsymbol{\mu}}{|\boldsymbol{r}|}) \end{aligned} \tag{4.7}$$

The term $\boldsymbol{\nabla}^2 \frac{\boldsymbol{\mu}}{|\boldsymbol{r}|}$ vanishes for $\boldsymbol{r} \neq 0$. Using equation (4.7) it is possible to write the dipolar field in terms of $1/r$ potentials, the handling of which is well-known from the literature. One just needs to apply the differential operators to every term of the expansion. In dipolar field FMM codes which make use of spherical harmonics the basis functions of the dipolar field can be expressed as derivatives of the spherical coordinates. As these form a complete set of basis functions the derivatives can again be expressed as spherical harmonics, which leads to an easy analytical expression for the expansion. Here, we focus on expanding a scalar Coulomb potential ϕ in Cartesian coordinates. Later on we show how this can be generalized to the vector magnetic potential $\boldsymbol{\phi}$.

To expand the Coulomb potential the vector $\boldsymbol{R}$ which points from the evaluation point to the particle will be separated into two vectors $\boldsymbol{r}$ and $\boldsymbol{r'}$ (see Fig. 4.9). The vector $\boldsymbol{r}$ is identified with the center of charge of a particle distribution.

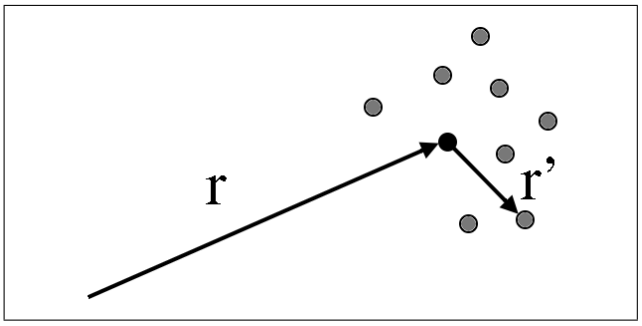


Figure 4.9: Distribution of particles around its center of charge at $\boldsymbol{r}$.

$$\phi = \frac{1}{|\boldsymbol{r} + \boldsymbol{r'}|} = \frac{1}{\sqrt{(x_1 + x_1')^2 + (x_2 + x_2')^2 + (x_3 + x_3')^2}}$$

Assuming that $\boldsymbol{r'} = (x_1', x_2', x_3')$ is small compared to $\boldsymbol{r} = (x_1, x_2, x_3)$, a Taylor expansion

around $\boldsymbol{r}$ can be used:

$$\begin{aligned}\phi &= \left(1+\sum_{\alpha=1}^{3} x'_\alpha \frac{\partial}{\partial x'_\alpha} + \frac{1}{2}\sum_{\alpha,\beta=1}^{3} x'_\alpha x'_\beta \frac{\partial^2}{\partial x'_\alpha x'_\beta}...\right)_{|_{r'=0}} \frac{1}{|\boldsymbol{r}+\boldsymbol{r'}|} \\ &= \frac{1}{|\boldsymbol{r}|} + \sum_{\alpha=1}^{3} x'_\alpha \frac{\partial}{\partial x_\alpha}\frac{1}{|\boldsymbol{r}|} + \frac{1}{2}\sum_{\alpha,\beta=1}^{3} x'_\alpha x'_\beta \frac{\partial^2}{\partial x_\alpha x_j}\frac{1}{|\boldsymbol{r}|}\end{aligned}$$

One can now introduce a system of charges q_i which are located around the vector $\boldsymbol{r}$. Expanding the system of charges around $\boldsymbol{r}$ leads to:

$$\begin{aligned}\sum_{n=1}^{N}\phi_n &= \left(\sum_{n=1}^{N} q_n\right)\frac{1}{|\boldsymbol{r}|} + \sum_{\alpha=1}^{3}\left(\sum_{n=1}^{N} q_n x_{n\alpha}\right)\frac{\partial}{\partial x_\alpha}\frac{1}{|\boldsymbol{r}|} + \frac{1}{2}\sum_{\alpha,\beta=1}^{3}\left(\sum_{n=1}^{N} q_n x'_{n\alpha}x'_{n\beta}\right)\frac{\partial^2}{\partial x_\alpha x_\beta}\frac{1}{|\boldsymbol{r}|} + ... \\ &= Qa_0 + \sum_{\alpha=1}^{3} D_\alpha a_\alpha + \frac{1}{2}\sum_{\alpha,\beta=1}^{3} Q_{\alpha\beta}a_{\alpha\beta} + ...\end{aligned}$$

For derivatives of the $1/r$ potential we use the following notation:

$$a_0 = \frac{1}{|\boldsymbol{r}|}, \quad a_\alpha = \frac{\partial}{\partial x_\alpha}\frac{1}{|\boldsymbol{r}|} \quad a_{\alpha\beta} = \frac{\partial^2}{\partial x_\alpha x_\beta}\frac{1}{|\boldsymbol{r}|}$$

The moments of the expansion up to the third order are listed below. The number of values increases by 3^l where l is the order of the expansion. This number can be reduced due to symmetry arguments.

$$Q = \sum_{n=1}^{N} q_n \qquad D_\alpha = \sum_{n=1}^{N} q_n x'_{n\alpha} \qquad Q_{\alpha\beta} = \sum_{n=1}^{N} q_n x'_{n\alpha}x'_{n\beta}$$

The same procedure can be used for expanding every component of the vector $\boldsymbol{\phi}$ in equation (4.6). A straight-forward calculation finally leads to:

$$\begin{aligned}\boldsymbol{\nabla\phi} &= \boldsymbol{\nabla}(\boldsymbol{Q'}a_0 + \sum_{\alpha=1}^{3}\boldsymbol{D'}_\alpha a_\alpha + \frac{1}{2}\sum_{\alpha,\beta=1}^{3}\boldsymbol{Q'}_{\alpha\beta}a_{\alpha\beta}) \\ &= \boldsymbol{Q'\nabla}a_0 + \sum_{\alpha=1}^{3}\boldsymbol{D'}_\alpha\boldsymbol{\nabla}a_\alpha + \frac{1}{2}\sum_{\alpha,\beta=1}^{3}\boldsymbol{Q'}_{\alpha\beta}\boldsymbol{\nabla}a_{\alpha\beta}\end{aligned}$$

where the multipole moments are given by:

$$\boldsymbol{Q'} = \sum_{n=1}^{N}\boldsymbol{\mu}_n \qquad \boldsymbol{D'}_\alpha = \sum_{n=1}^{N}\boldsymbol{\mu}_n x'_{n\alpha} \qquad \boldsymbol{Q'}_{\alpha\beta} = \sum_{n=1}^{N}\boldsymbol{\mu}_n x'_{n\alpha}x'_{n\beta}$$

These are marked with primes to distinguish them from the expansion of the Coulomb potential. The scalar values have now changed to vectors in which each component corresponds to the component of the magnetic moment $\boldsymbol{\mu}_n = (\mu^x, \mu^y, \mu^z)$.

The magnetic potential can now be transformed into a magnetic field using equation (4.7):

$$\begin{aligned} B_\gamma &= \partial_\gamma(\boldsymbol{Q'}\boldsymbol{\nabla}a_0 + \sum_{\alpha=1}^{3}\boldsymbol{D}'_\alpha\boldsymbol{\nabla}a_\alpha + \frac{1}{2}\sum_{\alpha,\beta=1}^{3}\boldsymbol{Q}'_{\alpha\beta}\boldsymbol{\nabla}a_{\alpha\beta}) \\ &= \boldsymbol{Q'}\boldsymbol{\nabla}a_\gamma + \sum_{\alpha=1}^{3}\boldsymbol{D}'_\alpha\boldsymbol{\nabla}a_{\alpha\gamma} + \frac{1}{2}\sum_{\alpha,\beta=1}^{3}\boldsymbol{Q}'_{\alpha\beta}\boldsymbol{\nabla}a_{\alpha\beta\gamma} \end{aligned} \tag{4.8}$$

This equation forms the basic expansion for the dipolar field used in the multipole method. The implementation has been done up to the order given in equation (4.8). As mentioned above some entries of the multipole tensor depend on each other by symmetry and, thus, not all need to be calculated.

4.5 Translation of the Origin of a Multipole

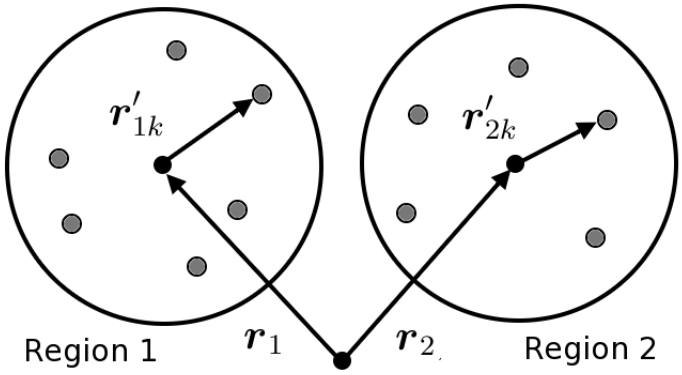


Figure 4.10: Subcells (Region 1 and 2) of a parent cell. Vectors $\boldsymbol{r_1}$ and $\boldsymbol{r_2}$ are pointing from the center of the parent cell to the center of the subcells. Vectors $\boldsymbol{r}'_{1k}$ and $\boldsymbol{r}'_{2k}$ are pointing from the centers of the subcells to each particle.

The origin of each multipole moments $\boldsymbol{Q}'_1 = Q'^{\alpha}_1$, $\boldsymbol{D}'^{\beta}_1 = D^{\alpha\beta}_1$ are given by the center of the corresponding subcell. Hence, in order to combine subcells to a parent cell one needs to shift the origin to the center of the parent cell (which will than be called $\bar{\boldsymbol{Q}}'_1$, $\bar{\boldsymbol{D}}'^{\beta}_1$). The procedure will be explained for these moments:

$$Q'^{\alpha}_1 = \sum_{k=1}^{N}\mu^{\alpha}_{1k} \qquad D'^{\alpha\beta}_1 = \sum_{k=1}^{N}\mu^{\alpha}_{1k}r'^{\beta}_{1k} \tag{4.9}$$

The first multipole moment is independent of the origin. Therefore, no shift of the origin is needed. The shifting vector from the center of the subcell to the center of the parent cell is given by $\boldsymbol{r}_1$. Hence, the second multipole moment of the subcell with a shifted origin will be given by:

$$\begin{aligned}\hat{D}_1'^{\alpha\beta} &= \sum_{k=1}^{N} \mu_{1k}^{\alpha}(r_{1k}'^{\beta} + r_1^{\beta}) \\ &= \sum_{k=1}^{N} \mu_{1k}^{\alpha} r_{1k}'^{\beta} + \sum_{k=1}^{N} \mu_{1k}^{\alpha} r_1^{\beta} \\ &= D'^{\alpha\beta}_1 + r_1^{\beta} \cdot Q_1'^{\alpha}\end{aligned}$$

The sums that appear correspond to the non-shifted first and second moment of the subcell and are, thus, already calculated. Therefore, the shifted multipole moments can easily be determined by a combination of non-shifted moments of the subcells.
The same method also works for the higher multipole moments. The values of the quadrupole moments at a shifted origin can be calculated as follows:

$$\hat{Q}_1'^{\alpha\beta\gamma} = Q_1'^{\alpha\beta\gamma} + D_1'^{\alpha\beta} \cdot r_1^{\gamma} + D_1'^{\alpha\gamma} \cdot r_1^{\beta} + r_1^{\beta} \cdot r_1^{\gamma} \cdot Q_1'^{\alpha}$$

The multipole moment of the parent cell is simply given by adding up the shifted multipole moments of the child cell. The correctness of this method has been verified by comparing the multipole moment of the root node with a direct evaluation

4.6 Timing and Accuracy

First of all we investigated the accuracy properties of the algorithm. The accuracy of a calculation was determined using a relative error of the form:

$$\epsilon^{\alpha} = \sqrt{\frac{\sum_{i=1}^{N}(B_{i\,\mathrm{tree}}^{\alpha} - B_{i\,\mathrm{acc}}^{\alpha})^2}{\sum_{i=1}^{N}(B_{i\,\mathrm{tree}}^{\alpha})^2}}, \qquad \alpha = x, y, z$$

Here $B_{i\,\mathrm{tree}}^{\alpha}$ represents the component α of the magnetic field calculated by the multipole method and $B_{i\,\mathrm{acc}}^{\alpha}$ is the result of exact brute force calculation. Different parameters can be varied for optimizing the accuracy. First of all the opening angle can be changed, which has a strong influence on the accuracy. The following table shows the cubic lattice calculation for different values of the opening angle $\theta = s/d$:

θ	accuracy ϵ
0.3	$8.61 \cdot 10^{-4}$
0.4	$2.73 \cdot 10^{-3}$
0.5	$5.62 \cdot 10^{-3}$
0.6	$1.18 \cdot 10^{-2}$
0.7	$2.33 \cdot 10^{-2}$

Low values of the opening angle lead to more a accurate calculations. However, they also lead to a dramatic increase in computational time and can break down the $N \log N$ scaling. The second parameter is the maximum number of particles in the leave box.

For a system of 10000 spins randomly distributed in a square with a maximum number of 20 spins per leave box and an opening angle of $\theta = 0.5$, an accuracy of 10^{-6} can be reached. For particles ordered on a two dimensional cubic lattice the accuracy is, however, lower. Accuracy values of 10^{-3} are achieved which is sufficiently for any kind of spin-dynamics simulation, since the statistical error of the perturbations due to the temperature bath will still be higher.

Besides, the evaluation of the global error, it is interesting to know how the accuracy is distributed among each individual spin. Therefore, the relative error per spin has been investigated, which is defined as:

$$r_i = \frac{B^{\alpha}_{i\,\mathrm{tree}} - B^{\alpha}_{i\,\mathrm{acc}}}{B^{\alpha}_{i\,\mathrm{tree}}}$$

The relative error has been determined for each individual single spin, which have been randomly distributed in a unit square. Fig. 4.11 shows the relative error in form of a histogram. A formation of a peak around zero can be observed. One can see that the statistical error (width of the peak) dominates the systematical error (peak position). A more detailed analysis shows that accurate calculations correspond an evaluation of a high magnetic field and inaccurate calculations to low magnetic field. Since the influence of low magnetic fields is negligible (compared to the influence of the exchange constant), the inaccuracy in the calculation of low magnetic field does not influence the correctness of a spin-dynamic simulation run.

As discussed above the computational time of a brute force calculation increases proportional to N^2 where N is the number of spins. For the Barnes and Hut tree algorithm we expect an increase proportional to $N \log N$. A comparison between the brute force calculation and the Barnes and Hut algorithm is shown in Fig. 4.12.

In the spin-dynamics approach the spin positions are fixed. Therefore, the tree and the interaction list only need to be built up once whereas the multipole moment calculation and the force summation have to be executed every time the spin direction will change. Therefore, for the determination of the simulation time, we neglected the tree built-up and interaction list routines that are only called once. The accuracy of the calculation is in the range of $\epsilon = 10^{-4}$ and the measurement points are linearly fitted (Fig. 4.12) in a log-log plot. The increase of the computational time for the brute force calculation has been determined to be proportional to $N^{1.9}$ which is close to expected N^2 law. For the multipole code an increase proportional to $N^{1.2}$ could be determined. For 128 spins the plot shows that the multipole code is already faster than a brute force calculation. Hence, the multipole code is for almost any practical application faster than a direct evaluation.

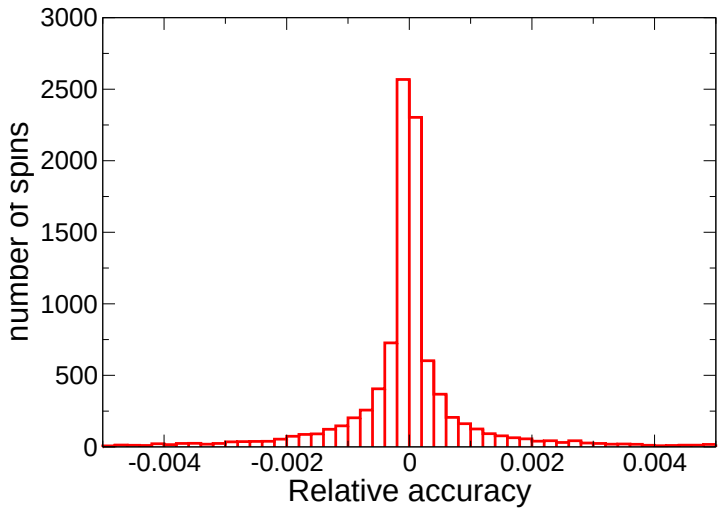


Figure 4.11: Relative accuracy of a multipole calculation of a randomly distributed cubic two dimensional spin system in a histogram.

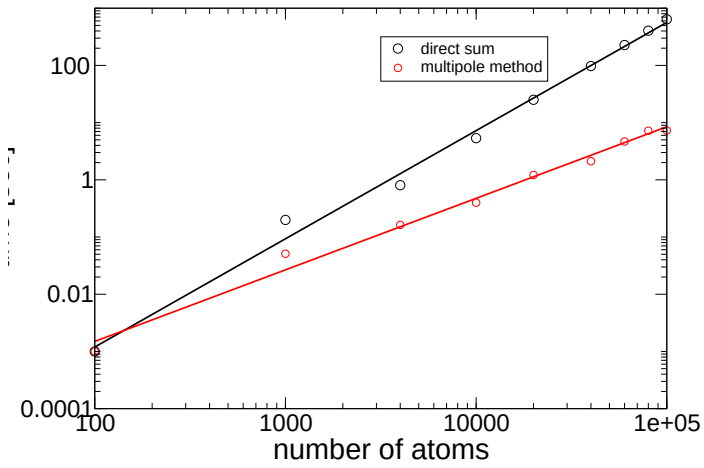


Figure 4.12: Log-log plot of timings for the evaluation of the magnetic field as a function of the number of particles in the system: (black) Brute force evaluation $\sim n^{1.9}$, (red) Multipole Method $\sim n^{1.2}$ using an Intel Core 2 CPU (2.13GHz).

5 Applications

5.1 Magnetization Curve of a 10x10x10 Cubic System

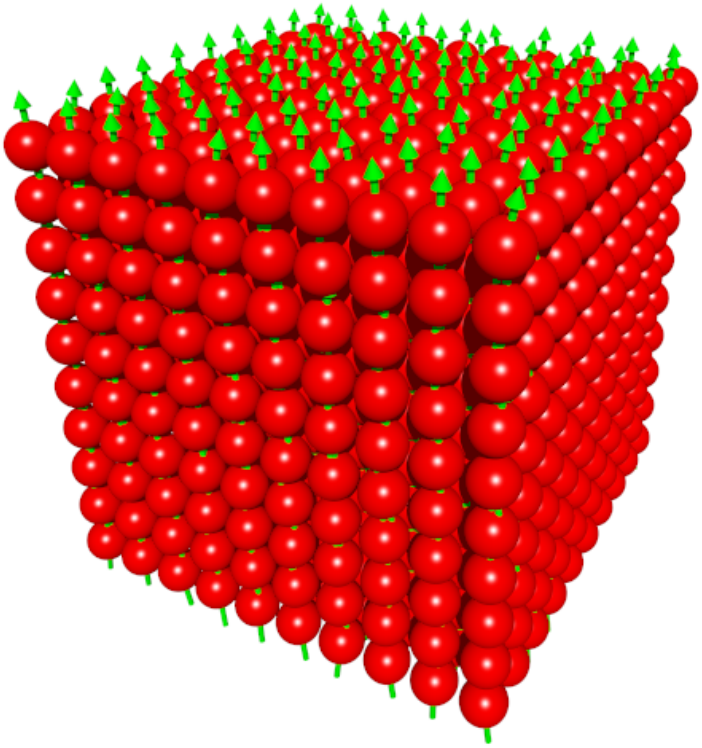


Figure 5.1: System of $10 \times 10 \times 10$ cubic lattice of atomic magnetic moments.

Introduction

The main purpose of the classical spin-dynamics approach is to investigate non-equilibrium magnetization dynamics. However, for a further testing of the method and for verifying the correctness of the code, we first study equilibrium properties. A lot of methods are commonly used for the determination of magnetization curves, which show the amplitude of the total magnetization $\boldsymbol{M}$ with respect to different temperatures. The most common are classical Monte Carlo methods which are here used for a comparison. There are also quantum mechanical methods, such as the random phase approximation (RPA)

[48], which determines magnetization curves in reasonable time.

Procedure

An infinite bulk system is simulated by $10 \times 10 \times 10$ atoms in a cubic lattice with periodic boundary conditions. The exchange constant for nearest neighbor atoms was chosen to be $J = 1\text{mRyd}$ which is a typical value for a bulk ferromagnet. The damping parameter for studying equilibrium quantities is arbitrary, which means different values will result in the same equilibrium properties. Nevertheless, it can be tuned to reduce computational time, as will be shown in this chapter.

For different values of temperature the initial state of the spin system is set to a ferromagnetic configuration. The simulation is then started and the normalized magnetization

$$M(t) = \frac{1}{N}|\boldsymbol{M}(t)| = \frac{1}{N}|\sum_i \boldsymbol{m}_i(t)| \tag{5.1}$$

is plotted as a function of time for $N = 1000$ atoms:

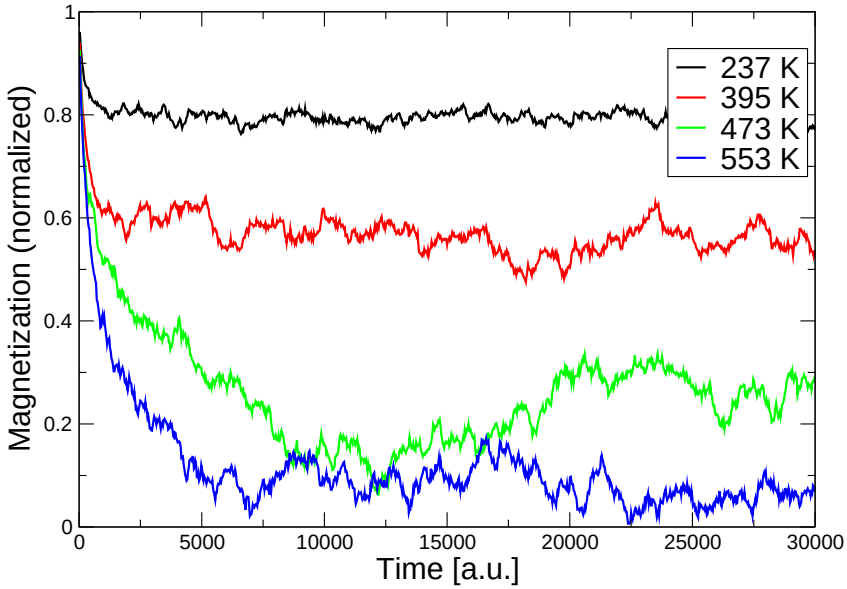


Figure 5.2: Magnetization of a $10 \times 10 \times 10$ atoms cubic lattice at different temperatures.

The normalized magnetization will, in time, converge towards a constant value, the equilibrium state. Then, it will fluctuate around the equilibrium state due to the influence of the temperature bath. The normalized magnetization values as a function of

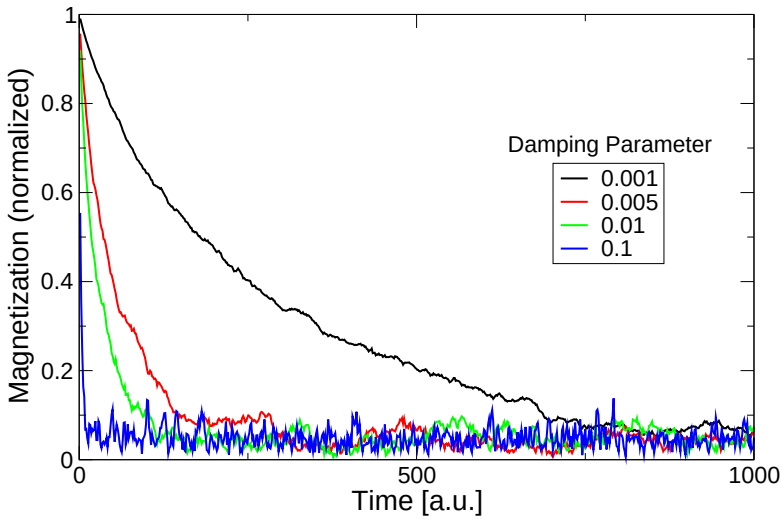


Figure 5.3: Magnetization of a $10 \times 10 \times 10$ atoms cubic lattice for different damping parameters at a 1.5 times the Curie temperature.

time can be seen in Fig. 5.2 for different temperatures. The magnetization values are determined by averaging in time after an equilibrium state is reached. Averaging is also used over different simulation runs with different sets of random numbers to reduce the statistical error. In this simulation run, an average over 15000 time units and 20 different sets of random numbers was used, which proved to be sufficient for the determination of the magnetization with a acceptable error.

As previously mentioned, the damping parameter does not influence the equilibrium properties. However, there is a dependence on the converging time, the time which is needed to reach an equilibrium magnetization. Figure 5.3 shows the normalized magnetization in time using a 1.5 times the Curie temperature for different values of the damping parameter. One can observe an antiproportional relation between the damping time and the converging time. The plot is shown in timescale of simulation units. The conversion factors are explained in the Appendix. A.2. The same behavior was also observed for temperatures below the Curie temperature. Thus, by increasing the damping parameter, the simulation time can remarkably be lowered. However, the numerical accuracy (see Appendix A) also decreases with the increase of the damping constant, which, therefore, needs to be carefully chosen.

Beside the magnetization the susceptibility has been analyzed for the bulk system. It determines the response of a system to an applied magnetic field $\boldsymbol{H}_{Zeeman}$ and is defined as:

$$\chi_{\mu\nu} = \frac{\partial M^{\mu}}{\partial H^{\nu}_{Zeeman}}|_{H_{Zeeman}=0}, \quad \mu, \nu = (x, y, z) \tag{5.2}$$

By virtue of the linear response theorem, this quantity can be determined without applying an external magnetic field.

Starting from a Hamiltonian of the form $\mathcal{H} = \mathcal{H}' + \sum_i \boldsymbol{m}_i \cdot \boldsymbol{H}_{\text{Zeeman}}$ where $\mathcal{H}'$ are contributions which are independent of the external magnetic field and $\boldsymbol{m}_i$ is the magnetic moment at site i, one can determine the average magnetization per atom using the Boltzmann statistics,

$$\langle M \rangle = \frac{1}{\sum_{\{\boldsymbol{m}\}} e^{-\beta\mathcal{H}(\{\boldsymbol{m}\})}} \sum_{\{\boldsymbol{m}\}} M e^{-\beta\mathcal{H}(\{\boldsymbol{m}\})}, \quad \beta = k_B T \tag{5.3}$$

where the sums run over all the possible 'microstates' $\{\boldsymbol{m}\} = \{\boldsymbol{m}_1, \boldsymbol{m}_2, ..\boldsymbol{m}_N\}$ and M is simply the magnitude of the total magnetic moment (equation (5.1)).

By inserting equation (5.3) into equation (5.2) one can derive a simple formula to determine the longitudinal susceptibility, which is, for the isotropic case, given by:

$$\chi = \frac{1}{k_B T}(\langle M^2 \rangle - \langle M \rangle^2)$$

Similar to the normalized magnetization the function χ will, in the equilibrium regime, fluctuate over time around a constant value. Hence, averaging will again be needed. Since the average runs over the second moments of the magnetization values, it leads to a higher statistical error compared to the normalized magnetization.

Analysis

The magnetization and susceptibility have been simulated for different temperatures leading to the magnetization curve in plot 5.4. The results have been compared to Monte Carlo simulations. One can see that the magnetization almost agrees perfectly. The susceptibility which depends on the magnetization fluctuations is slightly smaller than the results obtained by the Monte Carlo simulation. One reason is the higher statistical error for determining the susceptibility compared to the statistical error of the magnetization curve, as discussed previously. Nevertheless, the position of the peak of the susceptibility, signalling the Curie temperature, is in a very good agreement with the one obtained by the Monte Carlo method.

There are some limitations in the classical approximation, reflecting in both the Monte Carlo and the Spindynamics solutions. One can see that for small temperatures the magnetization decreases linearly in contradiction to the $T^{3/2}$ law of Bloch [49], which is related to the break-down of the classical approximation in the low temperature regime. One would also expect the magnetization to exactly drop to zero above the Curie temperature. However, the magnetization smoothly approaches zero for higher temperatures which can be explained by the finite size of the simulation supercell.

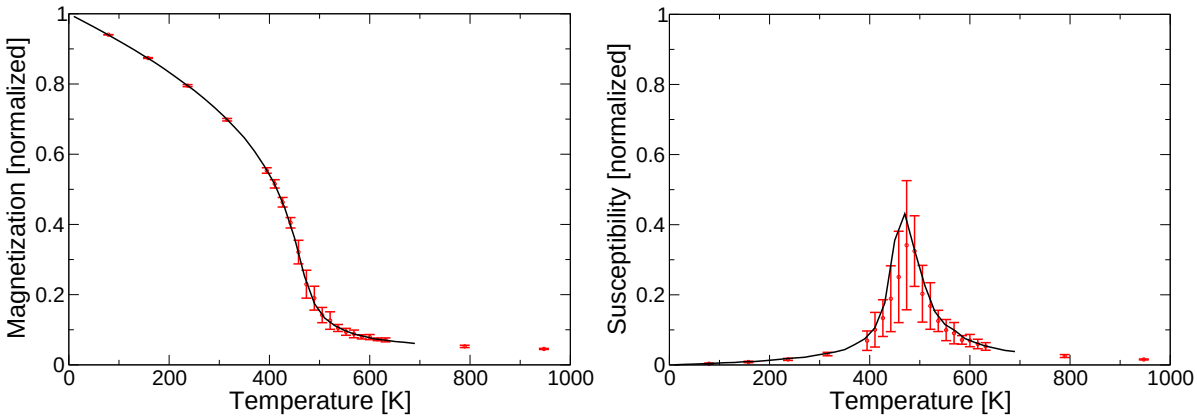


Figure 5.4: (left) Magnetization curve of a $10 \times 10 \times 10$ simple cube with periodic boundary conditions and (right) the susceptibility of the simple cube obtained by a spin-dynamics simulation (red) and compared to a Monte Carlo simulation (black).

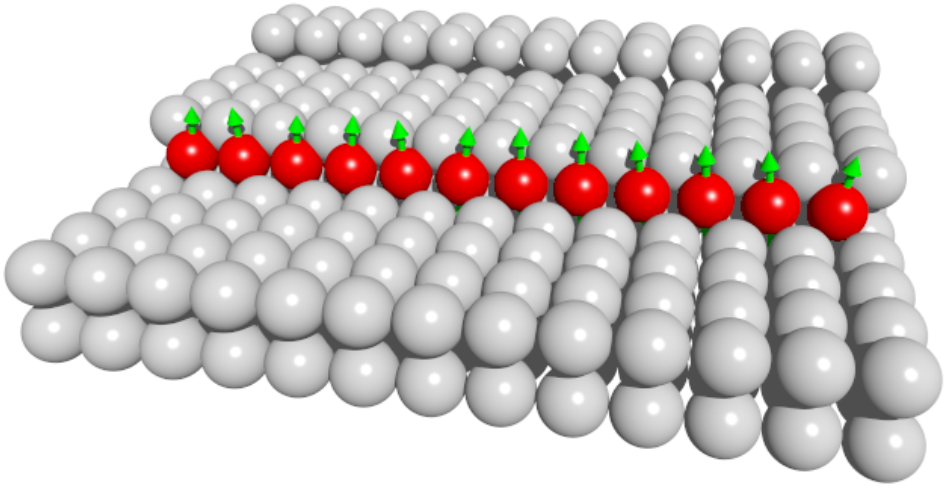


Figure 5.5: One dimensional chain of atomic moments on a surface.

5.2 Relaxation Times of One Dimensional Systems

Experimentalists were recently able to produce and study one dimensional chains on substrates. Gambarella et al [4] were able to grow and investigate the electronic properties and the ferromagnetism of Cobalt chains on Platinum. A picture of Cobalt chains deposed on Platinum can be seen in Figure 5.6. This gave us the motivation to study the spin behavior of finite but long chains considering periodic boundary conditions.

An important issue in one-dimensional systems is the stability of the ferromagnetic state.

In 1968, Mermin and Wagner [11] have proven that there is no spontaneous long-range

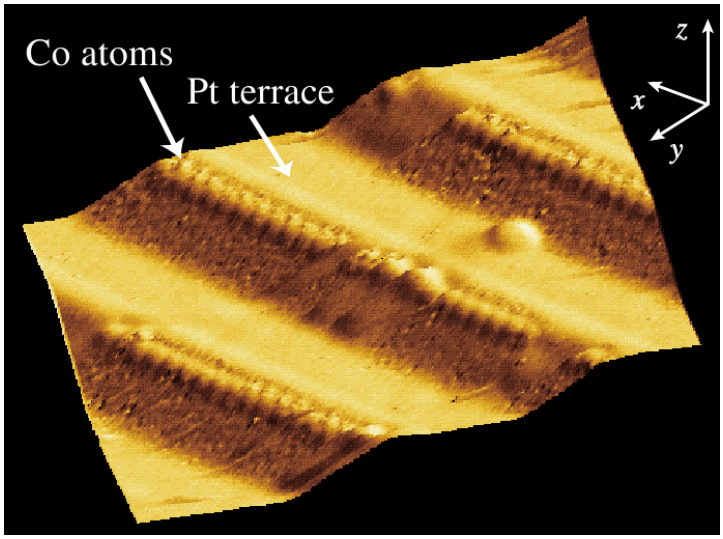


Figure 5.6: STM image of Cobalt chains which align on the step edges of a Platinum surface, picture taken from [4].

ferromagnetic order at $T > 0$ for isotropic one and two-dimensional Heisenberg models with short-range interaction. For a two-dimensional model, there can be a long-range order if magnetocrystalline anisotropies are present.

The difference between a one, two and three dimensional system can be understood by [50] analyzing the density of states of magnetic excitations. In the low energy approximation ($E \sim q^2$) the density of states for a d-dimensional system is:

$$D(E) \sim \begin{cases} E^{1/2}, d = 3 \\ \text{const}, d = 2 \\ E^{-1/2}, d = 1 \end{cases}$$

Thus, one can determine the change of the ground state magnetization at a finite temperature:

$$\Delta m \sim \int_0^\infty dE \frac{D(E)}{e^{E/k_B T} - 1} \quad \rightarrow \begin{cases} \infty & d = 1, 2 \\ \text{finite} & d = 3 \end{cases}$$

For any non-zero temperature the change of the ground state magnetization diverges in one and two-dimensions, which is a strong argument towards a loss of long range order. A detailed derivation of the Mermin-Wagner theorem can be found in [51, 52].

However, it is important to know what is the time scale in which an initially imposed ferromagnetic order is lost. This can be studied with the developed spin-dynamics simulation code. Figure (5.7) shows the total magnetization of chains with different length as a function of temperature. One can clearly observe a drop of the magnetization towards zero. This indeed happens for any finite temperature. Nevertheless, finite size

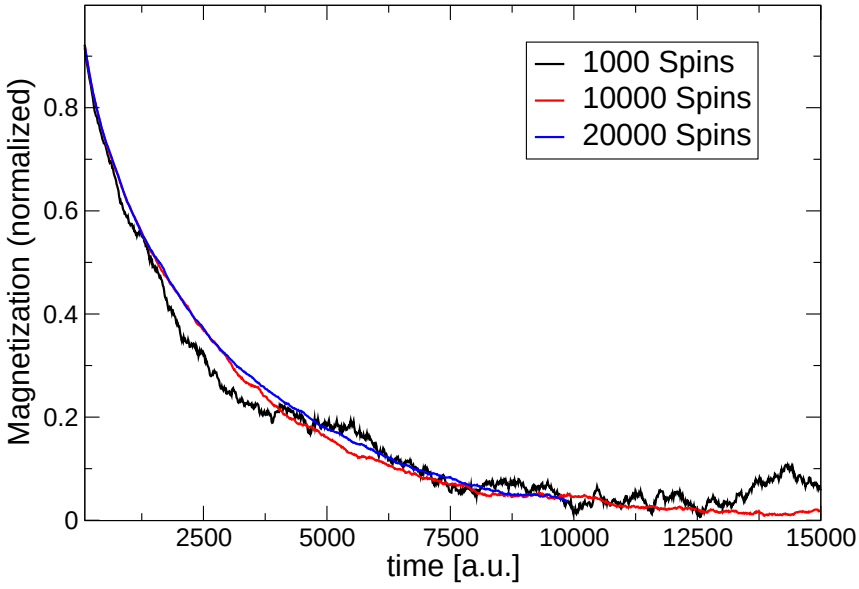


Figure 5.7: Magnetization of spin chains with different length as a function of temperature.

effects are observable. First, the drop of the magnetization is not smooth. Fluctuations are present as already observed for the magnetization of a bulk system. Their amplitude is proportional to $\frac{1}{\sqrt{n}}$ where n is the number of spins in the chain. Thus, smoother results can be obtained for systems of large chains.

For the simulation of an infinite chain, one needs to find a tradeoff between accurate calculations (low fluctuations) and saving computational time. The following calculations were done with chain length between 5000 and 10000 spins, where the error is acceptably small. An exponential drop of the magnetization with respect to the time is expected. This behavior could be proven in the simulation by using a logarithmic plot (Fig. 5.8). One can see that the exponential behavior ceases for the 1000 spins system in the time range above 7500 simulation time units. The crucial point in accuracy is reached as soon as the fluctuations of the magnetization dominate its total value. For huge spins systems the exponential regime is much broader than for small systems.

Next we investigate the demagnetization time, i.e. the time in which the magnetization drops to zero, as a function of the temperature. Considering a system with 5000 spins with a nearest neighbor coupling of $J = 1\text{mRyd}$, a damping parameter $\lambda = 0.001$ and different temperature values, the magnetization curves are fitted to an exponential function:

$$m(t) = m_0 \cdot e^{-t/\tau}$$

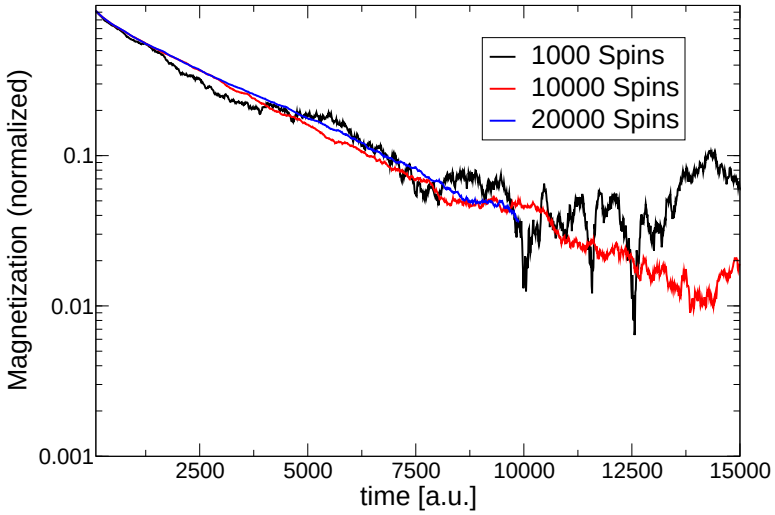


Figure 5.8: Logarithm of the magnetization of spin chains with different length as a function of temperature.

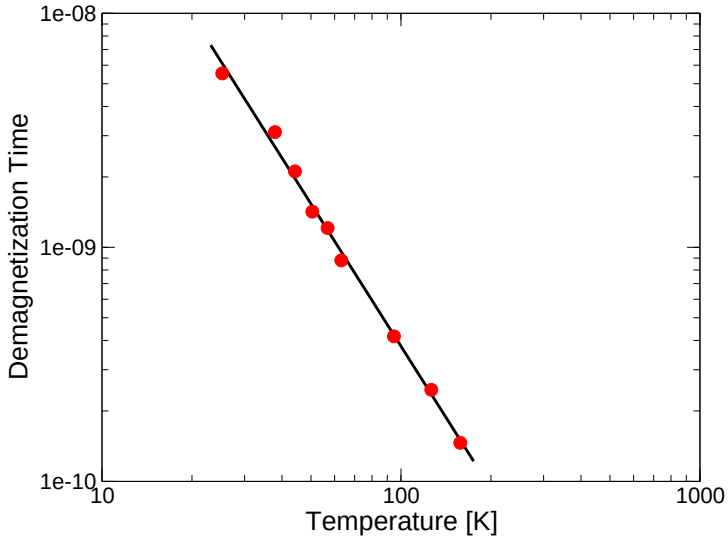


Figure 5.9: Demagnetization time τ of a 5000 spins chain as a function of temperature.

The factor in the exponent τ will futher on be referred to as the demagnetization time. The different values are shown as a log-log plot in Fig. 5.9 in the regime where the exponential behavior is valid. The linear behavior which can be observed in the plot

represents a power law with a negative exponent, which is reasonable since it is expected that the demagnetization time diverges at low T and is practically instantaneous at high T.

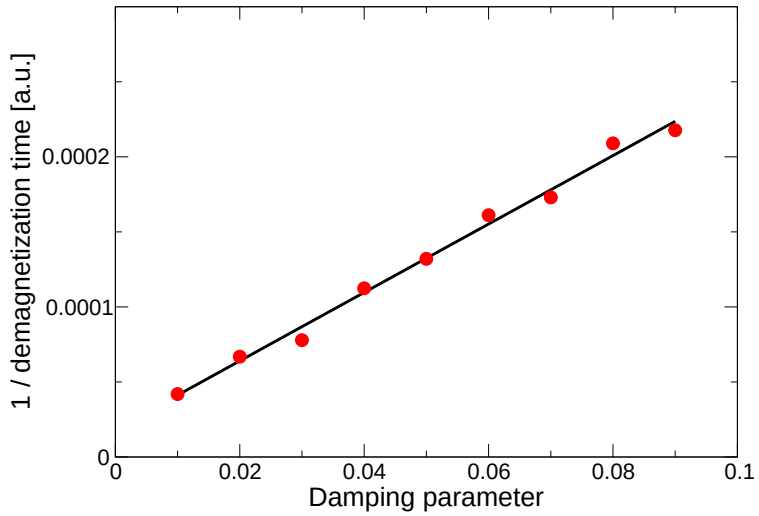


Figure 5.10: Demagnetization time τ of a 5000 spins chain as a function of the damping parameter λ.

After investigating the behavior of the demagnetization with respect to the temperature, the dependence on the damping parameter λ is investigated. Figure 5.2 shows the dependence of the inverse demagnetization time as a function of the damping parameter. A linear behavior can be observed. Thus, there is an antiproportional behavior between the demagnetization time and the damping parameter, which is in agreement with the results obtained for the bulk system in Chapter 5.1 (Fig. 5.3).

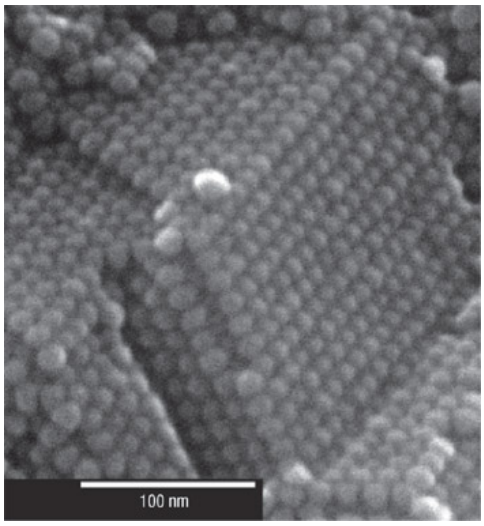


Figure 5.11: Formation of a FeCo nano-particle on a superlattice, picture taken from [53].

5.3 Switching Times of a Nano-Cluster on a Surface

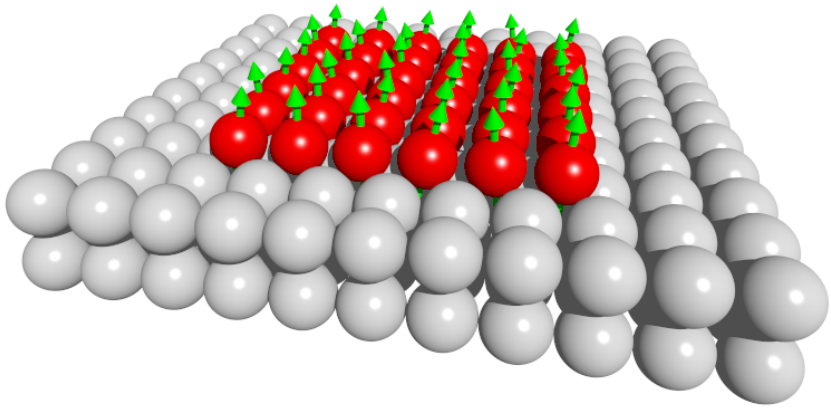


Figure 5.12: Nano-cluster of atomic moments on a surface.

Introduction

A key issue in future magnetic data storage is the magnetization reversal dynamics of finite magnetic particles. Magnetic discs are used to store information in form of bits by individually setting the magnetization of each block either 'up' or 'down'. In the past years there has been a huge increase of data capacity by reducing the size of each block.

However, blocks of reduced size soon become more sensitive to thermal fluctuations. Future promising candidates are magnetic nano-particles, in which a data density of

several terabits per square centimeter could be reached [1, 54]. As a comparison modern hard drives have data density of about 35 gigabit per square centimeter. As the magnetic anisotropy energy of each block approaches the thermal energy, the thermal effects will flip the magnetization randomly, a situation known as the superparamagnetic limit [55, 56]. Thus, the understanding of thermal stability of ferromagnetic nanostructures becomes important. The time until a thermally activated switch occurs varies over many orders of magnitude. It can be observed in the picoseconds regime up to the order of years. Therefore, different techniques need to be used to study different timescales. With spin-dynamics methods one is able to study ultra fast switching of nano-particles. Other techniques like the kinetic Monte Carlo [9] method are able to study switching behavior up to the time of seconds accurately, but fail in the description of ultra fast switching. Our interest mainly focuses on the thermal switching behavior of two-dimensional nano-clusters deposed on surfaces.

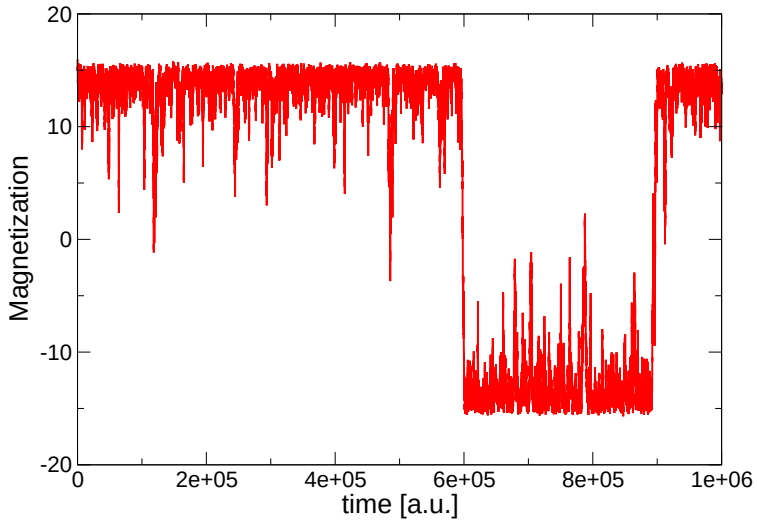


Figure 5.13: Projection of the magnetization of a 4×4 particle on the easy axis vs simulation time at $J = 1$, $K = 0.1J$,$T = 0.1J$.

We investigate the short-timescale magnetization dynamics of small nanoislands on a substrate due to thermal fluctuations. The presence of a substrate in the xy-plane is simulated by including an anisotropy term which has an easy axis towards the z-direction. Hence, two energy minima, one for all spins pointing parallel (spin-up state) and one for all spins pointing anti-parallel (spin-down state) to the unit vector $\boldsymbol{z}$, are obtained. Both states are energetically equivalent and, thus, have the same probability of being occupied. The spin system is expected to switch spontaneously between the spin-up state and the spin-down state, spending on the average an equal amount of time in each

state. Therefore, at non-zero temperature the average magnetization in the z-direction will be zero. We now proceed to investigate the switching behavior between the spin-up state and the spin-down state.

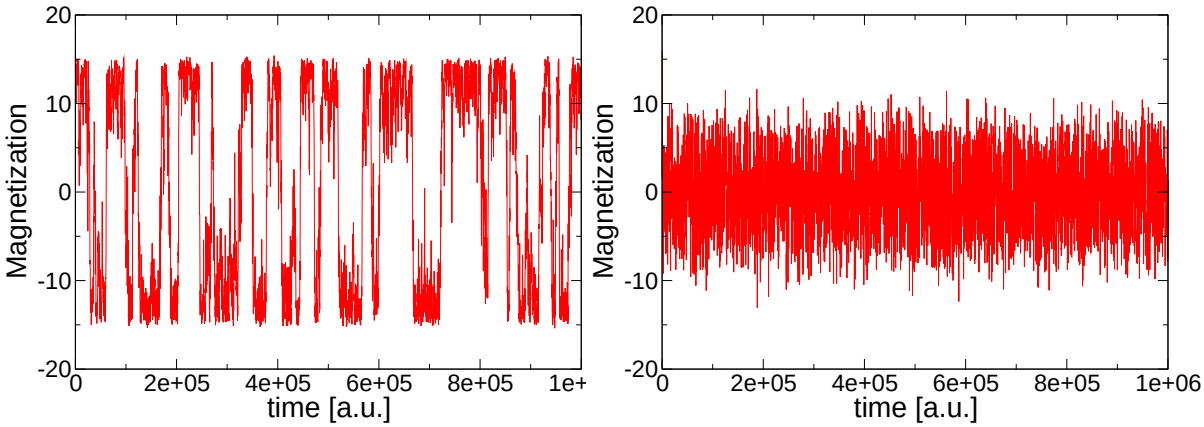


Figure 5.14: Magnetization switching of a 4×4 cluster with low anisotropy (K=0.04). The graph shows the projection of the magnetization on the easy axis for $T = 0.18J$ (left) and $T = 0.3J$(right). On the left hand sight one can see that the spin-up and the spin-down states almost not distinguishable, on the right hand sight one is not able to separate the two states.

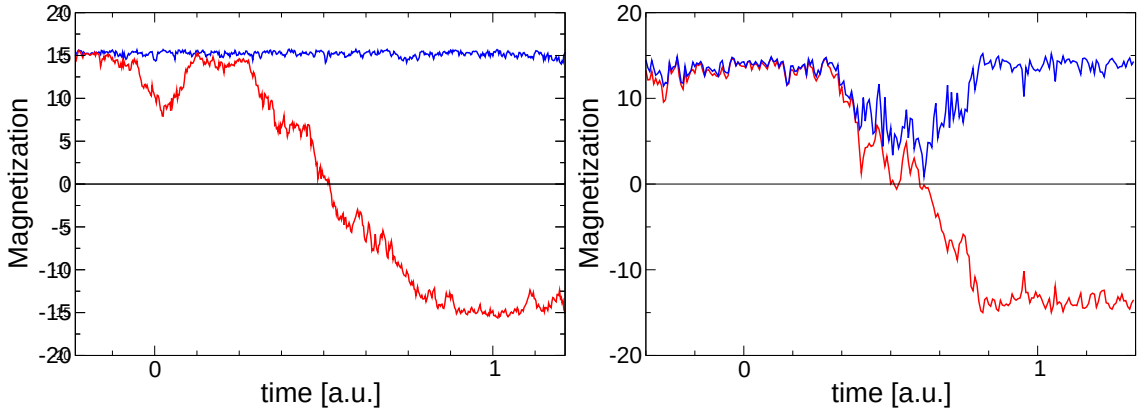


Figure 5.15: Magnetization switching of a 4×4 cluster during a switching event with low anisotropy ($K = 0.04J$,left) and high anisotropy ($K = 1.00J$, right). The red graph shows the total magnetization along the easy axis and the blue graph shows the length of the total magnetic moment.

A typical switching behavior of a particle can be seen in Fig. 5.13. At low temperatures, the z-component of the total spin stays almost constant around the ferromagnetic

ground state. If one increases the temperature the magnetization in the z-direction is lowered compared to the ground state but will still stay around a constant level. At a certain point a switching process occurs and the magnetization changes sign. Such switching processes occur continually over time. The average time between two switches is called the switching time.

The switching time is very sensitive to the parameters like temperature and anisotropy constant. For very high temperatures the thermal energy is of the order of the anisotropy energy or higher. Hence, the system does not remain at a particular up- or down-state for any appreciable amount of time. In Fig. 5.14 (left) the switching events are shown at a temperature for which one can hardly separate between the different states. For higher temperatures (Fig. 5.14 right) no spin-up or spin-down state can be formed. On the other hand, at sufficiently low temperatures the switching time increases to a point where no switching events can be observed for reasonable simulation times.

Types of Switching Mechanism

We first focus on the kind of switching behavior, which highly depends on the ratio between the exchange constant and the anisotropy constant (J/K-ratio). There are two limit cases: Systems with very low anisotropy and systems with high anisotropy.
A system with low anisotropy can be understood in terms of the Stoner-Wohlfarth model which describes the magnetization behavior of a finite particle in a single-domain state. For a system with a very low anisotropy, the Hamiltonian is, as an approximation, almost isotropic since the anisotropy energy is small compared to the exchange energy. Hence, one does not expect a different behavior for the magnitude of the total macro spin[1] in a spin-up, a spin-down or an in-between state. The individual spins will almost collinearly rotate around the anisotropy vector, which can be described by the Stoner single macro-spin model. The Stoner-Wohlfarth model gives reasonable results for low but fails for high anisotropy. The limiting case for high anisotropy leads to the well known Ising model in which just the spin-up or the spin-down state of a single spin can be occupied. For finite anisotropy, domains will be formed with a domain wall thickness proportional to $\sqrt{J/K}$ (see Appendix A.4). For very high anisotropies there will be a sharp separation between domains: In the Ising model one expects that at one instant in the switching event, there will be as many spins pointing up as down. Thus, at this particular instant the total moment will drop to zero.

The change in the magnitude of the total moment in a switching event can be monitored in simulations. Figure 5.15 shows in blue the length of the total moment switching for $K/J = 0.1$ (left) and $K/J = 1.0$ (right). On the left hand side the spins are strongly coupled. As discussed there is no drop down of the length of the total moment. On the right hand side a system with a high anisotropy is shown. One can observe a clear

[1] Sum of all single spins.

decrease of the magnitude of the macro spin during the switching process.

An anisotropy constant of the order of J is already a high value which is not found in bulk systems. But there are ways to design this property. It is, for example, possible to produce a system of magnetic atoms on surfaces in which the atoms are separated from each other by several Ångstrom [57, 58, 59]. The coupling constant will, thus, be reduced leading to a decrease of the J/K ratio.

For further analysis of the physics within a switching event the data of each individual spin has been written out for a 16×16 cluster. Snapshots of the spin state during a switching event of a 16×16 cluster system can be seen in Fig 5.16 and Fig 5.17.

Figure 5.16 shows the switching event of a cluster with high anisotropy ($J/K = 0.9$). The time is normalized to the switching time of the particle. Time 0 corresponds to the starting point of a switching event and time 1 corresponds to the end. The spin vectors are visualized by two spheres touching each other. Blue spheres represent the tails of arrows and red spheres the heads. The first and the last figure correspond to the spin-up (red sphere up) and spin-down (blue sphere up) states. In-between, a clear formation of a domain wall in the upper right corner can be seen, which propagates through the system until the whole cluster is in a spin down state. Sharp domains are visible, in which neighboring spins are aligned almost antiparallel.
Figure 5.17 on the other hand shows a cluster with a small anisotropy. It can be seen that neighboring spins are almost aligned in parallel. This again supports the argument that of a collinear rotation during a switch for low anisotropy which has been made in the discussion of Fig. 5.15. Nevertheless, a tendency to a formation of a domain wall is observable. In the figures (for $t = 0.49$, $t = 0.62$) one can observe a formation of a domain wall from the upper left to the lower right corner with a large domain wall, spread over the whole cluster.
The simulations show a clear trend to the limits previously discussed.

Switching times

In the simulations the system is defined to be in a spin-up state if the z-component of the total spin is bigger than 2/3 of its max. possible length, while it is considered to be in a spin-down state if the z-component of the total spin is less than $-2/3$ of its max. length:

$$\frac{1}{N}\sum_i S_i^z > +2/3 \cdot |\boldsymbol{S}| \qquad (\text{spin} - \text{up})$$

$$\frac{1}{N}\sum_i S_i^z < -2/3 \cdot |\boldsymbol{S}| \qquad (\text{spin} - \text{down})$$

Let us assume that a cluster reaches a spin-up state at time t_1 and then reaches a spin-down state at time t_2. Then the switching time of a single event is defined as $t_{sw} = t_2 - t_1$.

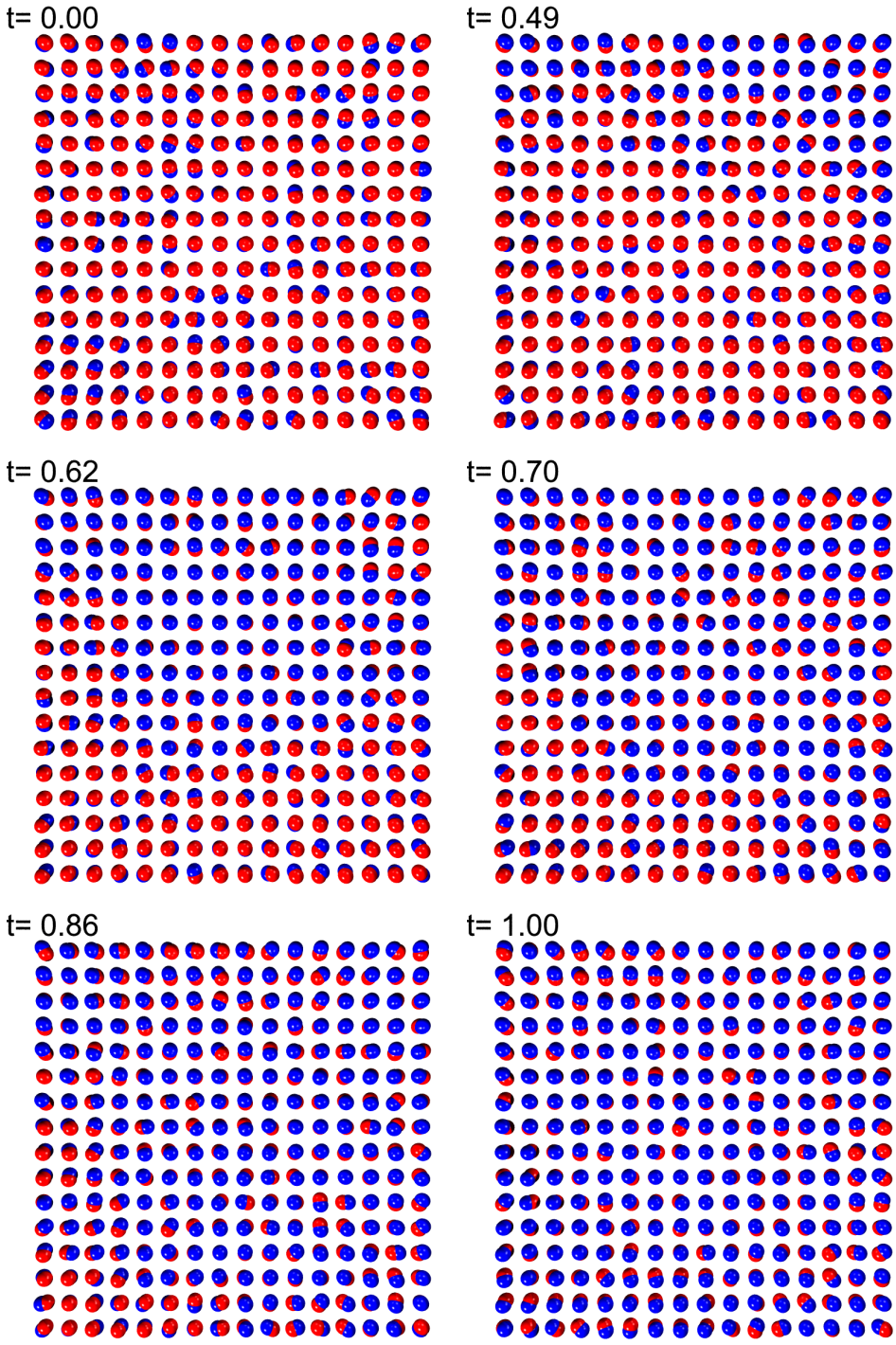


Figure 5.16: Snapshots of the spin state in a switching event for different relative switching times ($t = 0$ switching starts, $t = 1$ switching event ends) in a high anisotropy model.

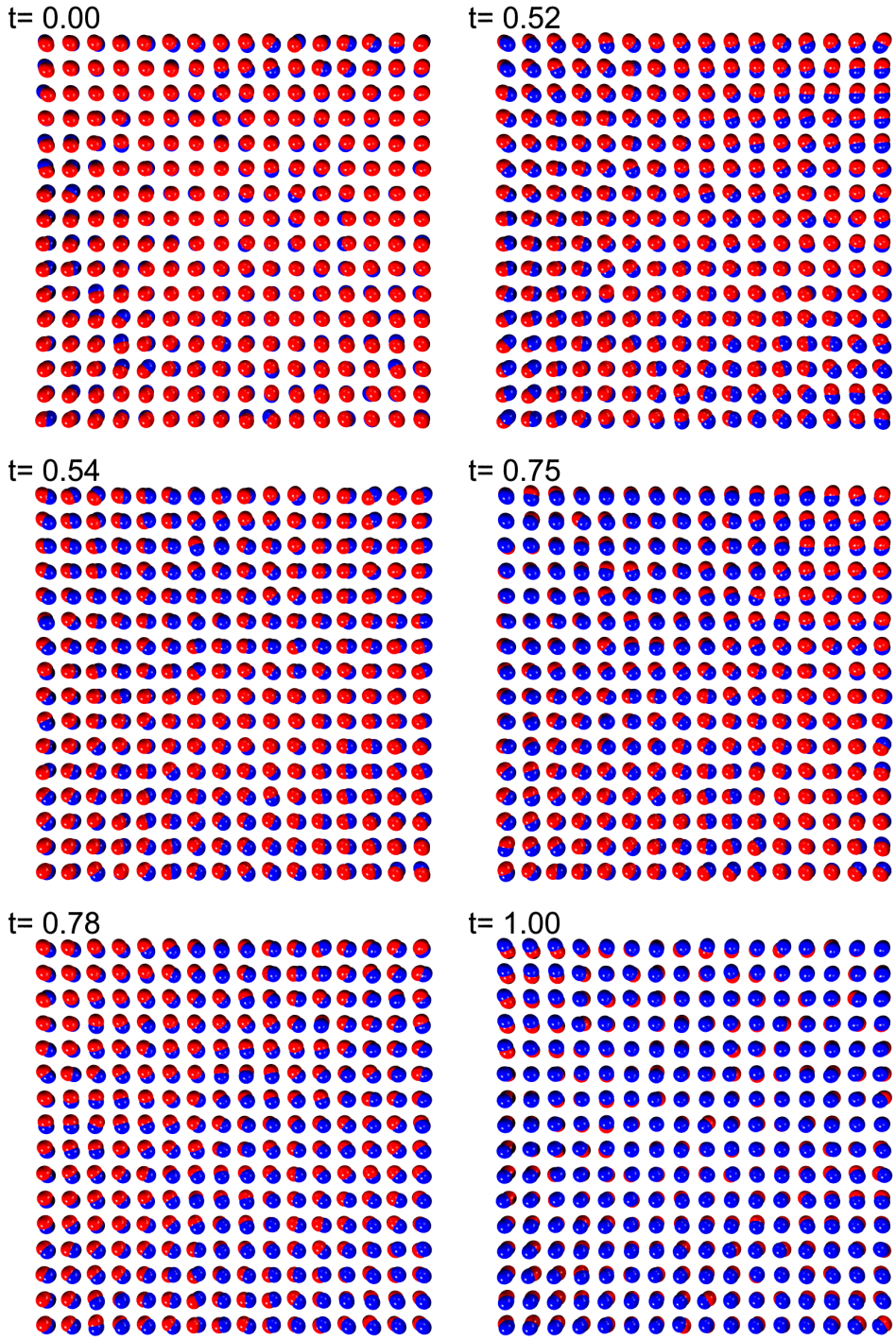


Figure 5.17: Snapshots of the spin state in a switching event for different relative switching times ($t = 0$ switching starts, $t = 1$ switching event ends) in a low anisotropy model.

Over long simulation time many switching events will take place with different switching times in form of a stochastic distribution. The distribution will depend on the parameters in the Hamiltonian, but its shape is always the same.
Switching events should have the following properties:

- the time between two switches is larger than the correlation time
- the probability of a spin switch should be invariant in time

This leads to a probability P of a switching event at time t_{sw} with an exponential distribution:

$$P(t_{sw}) = \frac{1}{\tau_s} \exp\left(-\frac{t_{sw}}{\tau_s}\right) \tag{5.4}$$

Here, τ_s is called the mean switching time (or for simplicity called the switching time) and denotes the mean value, as well as the standard deviation of the stochastic distribution. Quantitatively one can say that the probability of a switching event in a time frame of τ_s occurres with the probability of 63%. Numerically the switching time is determined by evaluation of the mean value of all switching times that occurred, which is a good approximation if one observes a large number of switching events. The numerical calculation of the standard deviation should lead to the same value as the mean value and can be used as a criterion to check if the amount of statistical data is enough.
The distribution (5.4) has been numerically verified for switching times above the correlation time. A histogram of the distribution of the switching time in a 4×4 cluster which supports our expectations can be seen in Fig. (5.18). The number of switching events are plotted in a logarithmic scale versus the switching times and a linear behavior is observable.
We expect a behavior for the switching time as follows:

$$\tau_s = \tau_0 \exp\left(\frac{\Delta E}{k_B T}\right) \tag{5.5}$$

where $1/\tau_0$ is often interpreted as an attempt frequency and ΔE corresponds to an energy barrier between degenerate spin-up and spin-down states. Equation (5.5) is also known as the Arrhenius Néel law [60].

To verify equation (5.5) we plotted the switching times of 4×4 particles with different anisotropy constants versus the inverse temperature. The linear behavior in the log-plot 5.19 shows that the model is applicable in the range of temperature and anisotropy constants used. The slope of each line corresponds to ΔE and the intersection between the exponential regression line with the y-axis corresponds to τ_0. Thus, one would expect the switching time to decrease toward τ_0 with increasing the temperature. However, this is not observed. Instead, there is, above some temperature, a breakdown of the spin-up spin-down picture. In this regime one is not able to determine a constant magnetization neither in the up nor in the down state as it is shown in Fig. 5.14(left). The spin will just fluctuate randomly between the two states (Fig. 5.14 right).

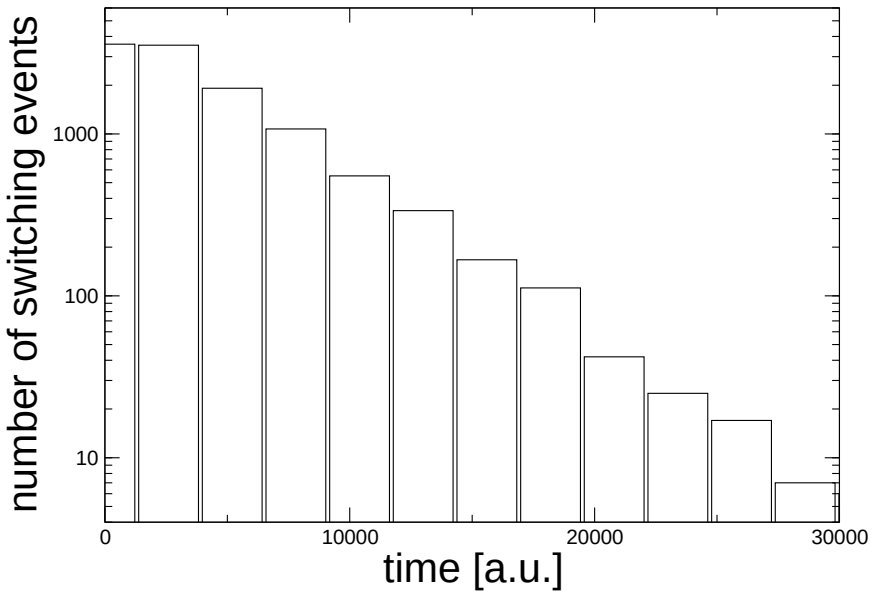


Figure 5.18: Histogram of the switching time of a 4×4 cluster with an anisotropy constant of $K = 0.1 \cdot J$.

As mentioned above, the behavior of the switching time follows equation (5.5) and the factor ΔE corresponds to the energy barrier between the spin-up and the spin-down state. In our case the energy barrier is given by the anisotropy at its highest value

$$E_K = S^2 \cdot N \cdot K$$

where S is the length of a single spin, N the number of spins and K is the anisotropy constant. A proportionality between ΔE and the anisotropy energy barrier E_K is expected. Hence, ΔE should be proportional to the number of atoms N and the anisotropy constant K.

To show this behavior calculations on different sized clusters (2×2, 3×3 and a 4×4) were performed. The slope of the curves of Fig. 5.19 is plotted versus the anisotropy constant as well as versus the number of atoms. Fig. 5.20 (left) supports our expectations and shows an almost linear behavior with respect to the anisotropy constant K. The error bars are given by the error due to the fit of the linear regression.

In Fig. 5.20 (right) one can also see a linear trend of ΔE for different particle sizes. In the regime of single domain particles the linear trend should be valid.

Concerning the behavior of bigger clusters, the computational time increases with increasing number of spins. Another problem is that the exponential factor in equation

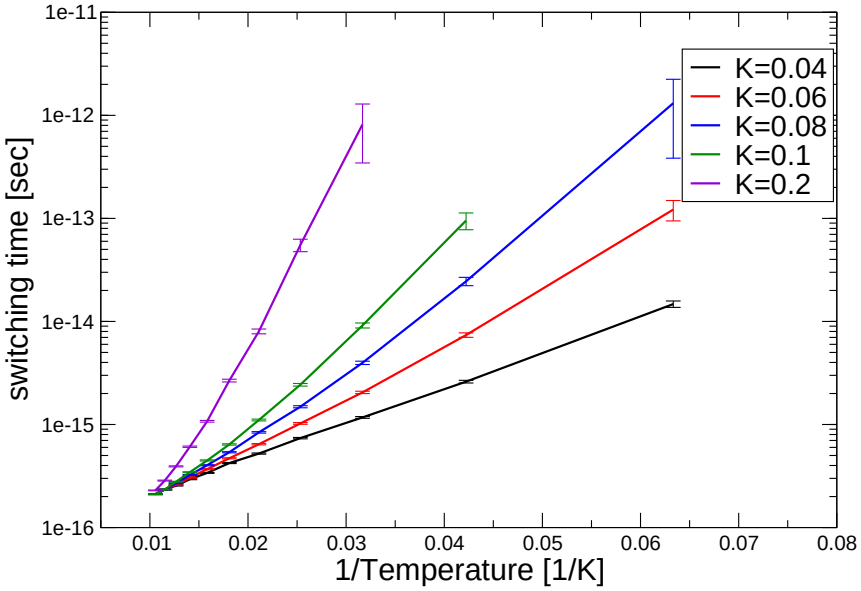


Figure 5.19: Switching times of a 4×4 particle with different anisotropy constants and an exchange constant of $J = 1\text{mRyd}$ versus the inverse temperature.

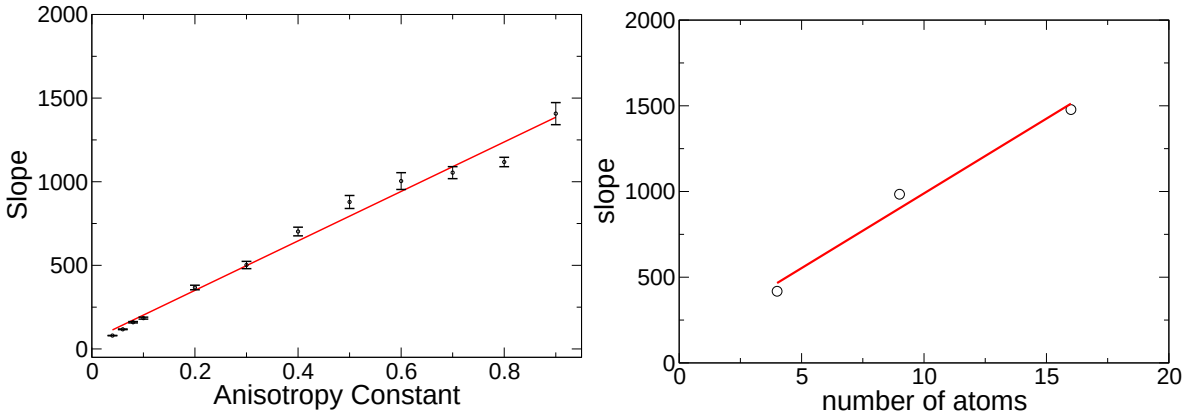


Figure 5.20: ΔE as a function of the anisotropy constant K (left) and the number of atoms N (right).

(5.5) will as well increase with the number of spins which leads to an exponential increase of the switching time. If one would like to scan the same temperature range used previously one would need to increase the simulation time exponentially with increasing number of spins. A future parallelization of the code could overcome this difficulty.

A Appendix A

A.1 Stochastic Integration Algorithm

A.1.1 Translation of the stochastic differential equation

In the following section the numerical algorithm used in this thesis for solving the following stochastic differential equation with multiplicative white noise will be discussed.

$$\frac{\partial \boldsymbol{S}_i}{\partial t} = \boldsymbol{h}_i \times \boldsymbol{S}_i - \lambda(\boldsymbol{h}_i \times \boldsymbol{S}_i) \times \boldsymbol{S}_i + \boldsymbol{f_i}(t) \times \boldsymbol{S}, \quad \boldsymbol{h}_i = \frac{\partial \mathcal{H}}{\partial \boldsymbol{S}_i} \tag{A.1}$$

The Hamiltonian $\mathcal{H}$ depends on the N different spin vectors $\{\boldsymbol{S}\} = \{\boldsymbol{S}_1, \boldsymbol{S}_2, ... \boldsymbol{S}_N\}$ and the random variables $\boldsymbol{f_i}(t)$ are given by a distribution of white noise. First, it is useful to rewrite the stochastic differential equation to an expression (see equation (A.2)), which is commonly used for the numerical analysis of stochastic differential equations. The differential equation is in the Ito interpretation of the integral and, therefore, needs to be transformed into the Stratonovich type, which is explained in section 2.4.2:

$$d\boldsymbol{S} = \boldsymbol{a}(\{\boldsymbol{S}\}) + \epsilon \sum_{r=1}^{3N} \boldsymbol{\sigma}_r(\{\boldsymbol{S}\}) \circ dW_r \tag{A.2}$$

$$= \boldsymbol{a}(\{\boldsymbol{S}\}) + \epsilon^2 \boldsymbol{b}(\{\boldsymbol{S}\})dt + \epsilon \sum_{r=1}^{3N} \boldsymbol{\sigma}_r(\{\boldsymbol{S}\})dW_r \tag{A.3}$$

For $\boldsymbol{a}, \boldsymbol{b}$ and $\boldsymbol{\sigma_r}$ the following expressions will be used:

$$a_i^\alpha = -A_i^\alpha - \lambda C_i^\alpha, \quad \boldsymbol{A}_i = \boldsymbol{s}_i \times \boldsymbol{h}_i, \quad \boldsymbol{C}_i = \boldsymbol{s}_i \times (\boldsymbol{s}_i \times \boldsymbol{h}_i)$$

$$b_i^\alpha(\boldsymbol{s}) = \frac{1}{2} \sum_{r=1}^{3N} \sum_{j=1}^{N} \sum_{\beta=x,y,z} \frac{\partial \sigma_r^{(i,\alpha)}(\boldsymbol{s})}{\partial \boldsymbol{s}_j^\beta} \sigma_r^{(i,\beta)}(\boldsymbol{s})$$

$$\sigma_{3i-1}^{(i,x)} = s_i^z, \quad \sigma_{3i}^{(i,x)} = -s_i^y, \quad \sigma_{3i}^{(i,y)} = s_i^x$$
$$\sigma_{3i-2}^{(i,y)} = -s_i^z, \quad \sigma_{3i-2}^{(i,z)} = s_i^y, \quad \sigma_{3i-1}^{(i,z)} = -s_i^x$$

The term $\boldsymbol{a}$ represents the driving force of the Hamiltonian, $\boldsymbol{b}$ is an expression which appears by transforming the Ito interpretation of the integral to the Stratonovich and $\boldsymbol{\sigma_r}$ is a sparse matrix which represents the crossproduct of the fluctuating term in equation (A.1).

A.1.2 Runge-Kutta Integration Scheme

Equation (A.3) is numerically solved by a weak Runge-Kutta (4th order) method for stochastic differential equations with small noise developed by Milstein and Tretyakov [35]. The method has a weak global error of $\mathcal{O}(h^4 + \epsilon^2 h^2)$ where $h = t_i - t_{i-1}$ denotes the timestep. By applying the Runge-Kutta algorithm to the equation of motion (A.3) one ends up with [18]:

$$\begin{aligned}
\boldsymbol{S}(t_{k+1}) &= \boldsymbol{S}_k + \epsilon h^{1/2} \sum_{r=1}^{3N} \boldsymbol{\omega}_r \xi_r + \epsilon^2 h \sum_{j,r=1}^{3N} \boldsymbol{c}_{j,r} \xi_{j,r} \\
&\quad + \frac{\epsilon h^{3/2}}{2} \sum_{r=1}^{3N} \boldsymbol{d}_r \xi_r + \frac{1}{6}[\boldsymbol{k}_1 + 2\boldsymbol{k}_2 + 2\boldsymbol{k}_3 + \boldsymbol{k}_4] \\
&\quad + \frac{\epsilon^2 h}{2} \Bigg[\boldsymbol{b} \cdot \boldsymbol{S}(t_k) + b\Bigg(\boldsymbol{S}(t_k) + \epsilon h^{1/2} \sum_{r=1}^{3N} \boldsymbol{\omega}_r \xi_r \\
&\quad + h[a(\boldsymbol{S}(t_k)) + \epsilon^2 \boldsymbol{b}(\boldsymbol{S}(t_k))] \Bigg) \Bigg]
\end{aligned} \tag{A.4}$$

The function $\boldsymbol{b}(\boldsymbol{s})$ is in our case simply given by $\boldsymbol{b}(\boldsymbol{s}) = -\boldsymbol{s}$ and the none zero components of c_{jr} and $\boldsymbol{d}_r$ are given by:

$$\begin{aligned}
c^{(i,x)}_{3i-2,3i-1} &= S^y_i(t_k) \quad , \quad c^{(i,x)}_{3i-2,3i} = S^z_i(t_k) \\
c^{(i,x)}_{3i-1,3i-1} &= -S^x_i(t_k) \quad , \quad c^{(i,x)}_{3i,3i} = -S^x_i(t_k) \\
c^{(i,y)}_{3i-1,3i-2} &= S^x_i(t_k) \quad , \quad c^{(i,y)}_{3i-1,3i} = S^z_i(t_k) \\
c^{(i,y)}_{3i-2,3i-2} &= -S^y_i(t_k) \quad , \quad c^{(i,y)}_{3i,3i} = -S^y_i(t_k) \\
c^{(i,z)}_{3i,3i-2} &= S^x_i(t_k) \quad , \quad c^{(i,z)}_{3i,3i-1} = S^y_i(t_k) \\
c^{(i,z)}_{3i-2,3i-2} &= -S^z_i(t_k) \quad , \quad c^{(i,z)}_{3i-1,3i-1} = -S^z_i(t_k)
\end{aligned} \tag{A.5}$$

$$\begin{aligned}
d^{(i,x)}_{3i-1} &= a^z_i(\boldsymbol{S}(t_k)) + \epsilon^2 b^z_i(\boldsymbol{S}(t_k)) \\
d^{(i,x)}_{3i} &= a^y_i(\boldsymbol{S}(t_k)) + \epsilon^2 b^y_i(\boldsymbol{S}(t_k)) \\
d^{(i,y)}_{3i} &= a^x_i(\boldsymbol{S}(t_k)) + \epsilon^2 b^x_i(\boldsymbol{S}(t_k)) \\
d^{(i,y)}_{3i-2} &= a^z_i(\boldsymbol{S}(t_k)) + \epsilon^2 b^z_i(\boldsymbol{S}(t_k)) \\
d^{(i,z)}_{3i-2} &= a^y_i(\boldsymbol{S}(t_k)) + \epsilon^2 b^y_i(\boldsymbol{S}(t_k)) \\
d^{(i,z)}_{3i-1} &= a^x_i(\boldsymbol{S}(t_k)) + \epsilon^2 b^x_i(\boldsymbol{S}(t_k)),
\end{aligned} \tag{A.6}$$

The random variables ξ_r and ζ_r are generated at each timestep according to the following stochastic distribution:

$$P(\xi = 0) = 2/3, \quad P(\xi = \sqrt{3}) = 1/6, \quad P(\xi = -\sqrt{3}) = 1/6 \tag{A.7}$$

$$P(\zeta = -1) = 1/2, \quad P(\zeta = 1) = 1/2$$
$$\xi_{jr} = 1/2(\xi_j \xi_r - \gamma_{jr}\zeta_j\zeta_r), \quad \gamma_{jr} = \begin{cases} -1, & j < r \\ +1, & j \geq r \end{cases}$$

$P(\xi = a)$ describes the probability of an event $\xi = a$. The stochastic distribution is (i.e. (A.7)) formed as follows: Three partitions are created which are assigned to a random event ($[0, 1/6] \rightarrow 1, [1/6, 1/3] \rightarrow -1, [1/3, 1] \rightarrow 0$). A random number generator which creates a uniform distribution in the range $[0, 1]$ is used. Depending on in which partition the random number occurs the different random events are assigned.

A.1.3 Pseudo-random number generator

The random number generator [61] uses the 'Mersenne Twister' algorithm [62], which was introduced by Makoto Matsumoto and Takuji Nishimura in 1997. It is important that random numbers are uncorrelated to each other and the period length in which the random numbers will be repeated is higher than the amount of random numbers used in a simulation run. The period length for the 'Mersenne Twister' algorithm is extremely long ($2^{19937} - 1 \approx 4 \cdot 10^{6001}$) and the typical amount of random numbers used can easily be estimated. A system of $100,000$ spins over 10^{12} timesteps will use $\sim 10^{17}$ random numbers which is far below the limit of $\sim 10^{6001}$.

A.2 Units Conversion

Time

The exchange constant J mostly governs the precession frequencies of each spin, which can be used to roughly estimate the precision of the numerical integration scheme with a constant timestep h. To keep an (almost) equal accuracy of the numerical integration scheme independent of the parameters in the Hamiltonian, the equation of motion is normalized to a system with a constant coupling parameter around 1. This leads to a rescaling of time, temperature and other parameters in the Hamiltonian. Let us assume an equation of motion without temperature of the following kind:

$$\hbar \frac{\partial \boldsymbol{S}}{\partial t} = \boldsymbol{S} \times \frac{\partial H}{\partial \boldsymbol{S}}, \quad H = \sum_{i=1}^{N} -g\mu_B H_{\text{Zeeman}} - K(e_z \cdot \boldsymbol{S})^2 + \sum_{i \neq j} -\frac{1}{2} J_{ij} \boldsymbol{S}_i \cdot \boldsymbol{S}_j$$

Dividing this equation by the highest exchange constant which is used during the calculation ($J = \max(J_{ij})$) will lead to:

$$\frac{\partial \boldsymbol{S}}{\partial \underbrace{(\frac{J \cdot t}{\hbar})}_{t'}} = \boldsymbol{S} \times \frac{\partial H}{\partial \boldsymbol{S}}, \quad H = \sum_{i=1}^{N} -g\mu_B \underbrace{\frac{H_{\text{Zeeman}}}{J}}_{H'_{\text{Zeeman}}} - \underbrace{\frac{K}{J}}_{K'} (e_z \cdot \boldsymbol{S})^2 + \sum_{i \neq j} -\frac{1}{2} \frac{J_{ij}}{J} \boldsymbol{S}_i \cdot \boldsymbol{S}_j \quad \text{(A.8)}$$

This expression can be simplified by introducing rescaled parameters, which are indicated by primed variables. They are used in each simulation run and will afterwards be transformed back. The fictive time t' can, for example, be transformed to the real time t according to:

$$\begin{aligned} t &= \hbar \cdot \frac{1}{J} \cdot t' = 1.054 \cdot 10^{-34} \cdot 4.587 \cdot 10^{17} \frac{1}{J}[\mathrm{Ry}] \cdot t' \mathrm{sec} \\ &= (4.837 \cdot 10^{-17} \frac{1}{J[\mathrm{Ry}]} \cdot t') \mathrm{sec} \\ &= (4.837 \cdot 10^{-14} \frac{1}{J[\mathrm{mRy}]} \cdot t') \mathrm{sec} \end{aligned}$$

Here, $J[\mathrm{Ry}]$ indicates unit-less numerical value of the exchange constant in Rydberg.

Temperature

In a heat bath model the temperature can be transformed in an analogous way to equation (A.8):

$$T' = k_B \cdot T/J$$

The primed parameters are in general dimensionless and are sometimes used in figures (if the exact values is of minor importance), which are indicated as arbitrary units ([a.u.]). The temperature conversion in terms of the exchange constant in units of Rydberg is given by the following expression:

$$T = \frac{J}{k_B} \cdot T' = 157.2463\mathrm{K} \cdot J[\mathrm{mRyd}] \cdot T'$$

A.3 Amdahl's law

The behavior of the computational time depending on the number of processors is often shown using the speedup of the parallel code which is defined as:

$$S_N = t_1/t_N$$

In the ideal case, while using N processors, a decrease of the computational time t_N by a factor N is expected. Hence, a perfect parallelization would lead to a speedup $S_N = N$. But even if there is no communication between processors there can be substantial speedup loss if the code cannot fully be parallelized. A single-processor code in which a part $f \cdot t_1$ (t_1 is the total time on a single processor) can be parallelized and $(1-f) \cdot t_1$ cannot, will have a running time on N processors of at least:

$$t_N = f \cdot t_1 + (1-f) \cdot t_1/N$$

Hence, the speedup of the code will result in :

$$S_N = t_1/t_N = \frac{1}{f + (1-f)/N}$$

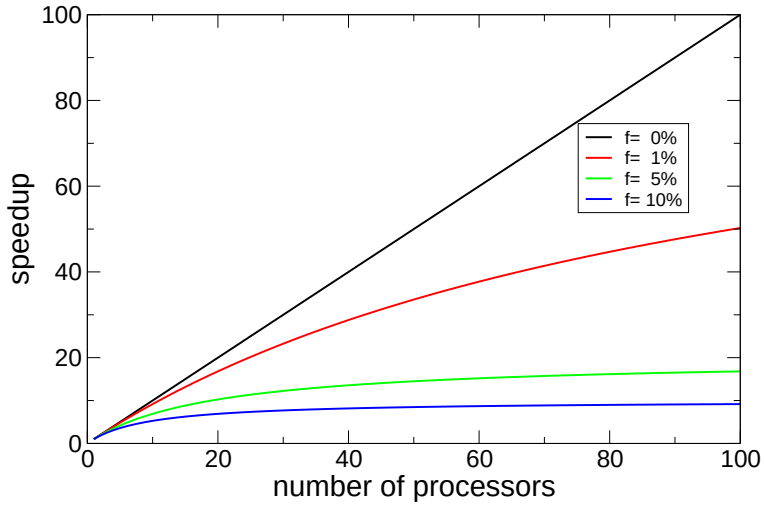


Figure A.1: Speedup of a code in which $(1-f)$ percent is parallelizable as a function of the number of processors (prediction by Amdahl's law).

A plot for different values of f in given in Fig. A.1. It is observeable that the speedup curves are substantially lowered for an increased non-parallelizable part of the code which converge to $1/f$ for high processor numbers. Thus, independent on the number of processors which are used the speed-up is limited by one over the non-parallelizable part of the program. In the following table typical values for f and the maximum speed-up are shown:

f	limit of the speedup
0.1%	$S_N < 1000$
1%	$S_N < 100$
5%	$S_N < 20$
10%	$S_N < 100$

A.4 Domain Walls formed by the Magnetocrystalline Anisotropy

Here, we examine the influence of the magnetocrystalline anisotropy and the exchange interaction on the width of magnetic domain walls.

The exchange interaction favors collinarity of neighboring spins leading to an increase of the domain wall width, whereas the anisotropy favors an alignment of the spins towards the easy axis leading to a decrease of the domain wall width. To explain [16] quantitatively the behavior of the domain wall thickness it is useful to transform the discrete atomistic model to a continuous one. The difference of angles of neighboring spins θ_i (compared to a reference vector) can then be approximated by the derivative

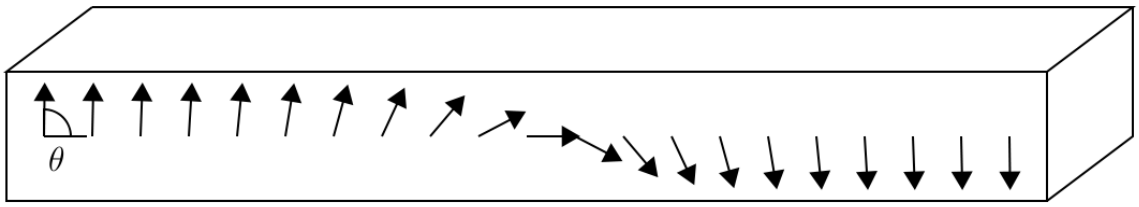


Figure A.2: Formation of a domain wall due to a magnetocrystalline anisotropy and the exchange coupling according to equation (A.9).

of the continuous function $\theta(x)$. Thus, the energy of an infinite chain can be written as follows:

$$E \sim \int_{-\infty}^{\infty} dx \left(A\dot{\theta}(x)^2 + K\cos^2\theta(x) \right)$$

Here, A is known as the spin-stiffness and is related to the exchange coupling J. Minimization of the function $\theta(x)$ with fixed boundary conditions at the borders leads to the following equation:

$$\theta(x) = 2\arctan\left(\exp\left(\frac{x}{\omega}\right)\right) - \frac{\pi}{2}, \qquad \omega = 2\sqrt{\frac{A}{K}} \tag{A.9}$$

Hence, the domain wall width is given by $\omega = 2\sqrt{\frac{A}{K}}$ which is in agreement with the previous qualitative argumentation. The domain wall will, especially in reduced dimension[1] (less nearest neighbors), be much smaller compared to a bulk system. For a layer of iron on W(110) a domain wall width of 0.6nm was measured, whereas already a double layer of iron increases the domain wall to 3.8nm. Domain walls in bulk are of the order of a hundred nanometers.

A.5 Implementation of the Multipole Method

The dipolar field is given by:

$$\boldsymbol{B}(\boldsymbol{r}) = \frac{3\boldsymbol{r}(\boldsymbol{r}\cdot\boldsymbol{\mu}_1) - \boldsymbol{\mu}_1\cdot r^2}{r^5},$$

One can define a magnetic potential which is similar to a Coulomb potential. The only difference is a substitution of the scalar charge with a magnetic moment vector. The dipolar field can then be written as:

$$\boldsymbol{B} = \boldsymbol{\nabla}(\boldsymbol{\nabla}\frac{\boldsymbol{\mu}}{r}), \quad \boldsymbol{\phi} = \frac{\boldsymbol{\mu}}{r} \tag{A.10}$$

[1] small A (less nearest neighbors), large K (low dimension)

An expansion of the magnetic potential up to the second order including the divergence operator results in:

$$\boldsymbol{\nabla\phi} = \boldsymbol{Q}'\boldsymbol{\nabla}a_0 + \sum_{\alpha=1}^{3} \boldsymbol{D}'_\alpha \boldsymbol{\nabla}a_\alpha + \frac{1}{2}\sum_{\alpha,\beta=1}^{3} \boldsymbol{Q}'_{\alpha\beta}\boldsymbol{\nabla}a_{\alpha\beta}$$

where the expansion coefficients are defined as:

$$\boldsymbol{Q}' = \sum_{n=1}^{N} \boldsymbol{\mu}_n \qquad \boldsymbol{D}'_\alpha = \sum_{n=1}^{N} \boldsymbol{\mu}_n x'_{n\alpha} \qquad \boldsymbol{Q}_{\alpha\beta} = \sum_{n=1}^{N} \boldsymbol{\mu}_n x'_{n\alpha} x'_{n\beta}$$

However, not all of the coefficients have to be calculated, because some are not independent from each other. The tensor $\boldsymbol{Q}_{\alpha\beta} = (Q_{x\alpha\beta}, Q_{y\alpha\beta}, Q_{z\alpha\beta})$ is, for example, symmetric in the sense that $\boldsymbol{Q}_{\alpha\beta} = \boldsymbol{Q}_{\beta\alpha}$. Thus, just the upper triangle of the tensor has to be calculated.

The expansion of the dipolar field used in the multipole code is given by the following equation:

$$\boldsymbol{B} = \boldsymbol{\nabla}(\boldsymbol{\nabla\phi}) = \boldsymbol{\nabla}U, \qquad U = \boldsymbol{\nabla\phi} = U_1 + U_2 + 0.5 \cdot U_3 + ...$$

Here, the potentials U_1, U_2, U_3 for each expansion order are given by a straight forward calculation

$$\begin{aligned}
U_1 \quad &= q_x \cdot a_x + q_y \cdot a_y + q_z \cdot a_z \\
U_2 \quad &= D'_{x1} \cdot a_{xx} + D'_{y2} \cdot a_{yy} + D'_{z3} \cdot a_{zz} \\
&\quad +(D'_{x2} + D'_{y1}) \cdot a_{xy} + (D'_{y3} + D'_{z2}) \cdot a_{yz} + (D'_{x3} + D'_{z1}) \cdot a_{xz} \\
U_3 \quad &= Q'_{x11} \cdot a_{xxx} + Q'_{y22} \cdot a_{yyy} + Q'_{z33} \cdot a_{zzz} \\
&\quad +(Q'_{y11} + 2Q'_{x12}) \cdot a_{xxy} + (Q'_{z11} + 2Q'_{x13}) \cdot a_{xxz} \\
&\quad +(Q'_{x22} + 2Q'_{y12}) \cdot a_{xyy} + (Q'_{z22} + 2Q'_{y23}) \cdot a_{yyz} \\
&\quad +(Q'_{x33} + 2Q'_{z13}) \cdot a_{xzz} + (Q'_{y33} + 2Q'_{z23}) \cdot a_{yzz} \\
&\quad +(2 \cdot Q'_{x23} + 2 \cdot Q'_{y13} + 2 \cdot Q'_{z12}) \cdot a_{xyz}
\end{aligned}$$

where the notation for derivatives of $1/r$ potential is given by:

$$a_0 = \frac{1}{|\boldsymbol{r}|}, \quad a_\alpha = \frac{\partial}{\partial x_\alpha}\frac{1}{|\boldsymbol{r}|} \quad a_{\alpha\beta} = \frac{\partial^2}{\partial x_\alpha x_\beta}\frac{1}{|\boldsymbol{r}|}$$

The expressions in the brackets (like $\bar{D}_{xy} = D_{x2} + D_{y1}$ or $\bar{Q}_{xxy} = Q_{y11} + 2Q_{x12}$) are combined to a single expansion coefficients leading to in total 3, 6 and 10 multipole moments for the first three expansion orders.

A.6 Dispersion relation

A.6.1 Two dimensional hexagonal lattice

The exact analytical solution for a spinwave in an infinite system with an arbitrary Bravais lattice structure is presented in chapter 3:

$$\hbar\omega = g\mu_B H_{\text{Zeeman}} + KS_z + JS_z(n - D(\boldsymbol{k}))$$

It depends on a quantity $D(\boldsymbol{k})$ which is defined as:

$$D(\boldsymbol{k}) = \sum_{\boldsymbol{\delta}} e^{\mathrm{i}\boldsymbol{k}\boldsymbol{\delta}} \tag{A.11}$$

where $\boldsymbol{\delta}$ runs over all the displacement vectors to the nearest neighbor atoms. The dispersion relation of a two dimensional hexagonal lattice structure is used to evaluate its density of states and, therefore, needs to be calculated. In this case, there are 6 displacement vectors with a nearest neighbor distance of a

$$\delta_{1,2} = (\pm a, 0), \quad \delta_{3-6} = (\pm 0.5\, a, \pm\sqrt{3}/2\, a)$$

which are inserted in equation (A.11):

$$\begin{aligned} D(\boldsymbol{k}) &= e^{\mathrm{i}k_x a} + e^{-\mathrm{i}k_x a} + e^{\mathrm{i}(\frac{1}{2}k_x a + \frac{\sqrt{3}}{2}k_y a)} \qquad (A.12)\\ &\quad + e^{\mathrm{i}(\frac{1}{2}k_x a - \frac{\sqrt{3}}{2}k_y a)} + e^{\mathrm{i}(-\frac{1}{2}k_x a + \frac{\sqrt{3}}{2}k_y a)} + e^{\mathrm{i}(-\frac{1}{2}k_x a - \frac{\sqrt{3}}{2}k_y a)} \\ &= 2\cos(k_x a) + 4\cos(\frac{\sqrt{3}}{2}k_y a)\cos(\frac{1}{2}k_x a) \qquad (A.13) \end{aligned}$$

Thus, the dispersion relation for a hexagonal two dimensional lattice is given by:

$$\hbar\omega = g\mu_B H_{\text{Zeeman}} + KS_z + JS_z\left(6 - 2\cos(k_x a) - 4\cos(\frac{\sqrt{3}}{2}k_y a)\cos(\frac{1}{2}k_x a)\right)$$

A.6.2 Two dimensional tetragonal lattice

The dispersion for a two dimensional cubic lattice can be calculated analog to the hexagonal case lattice using the four displacement vectors $\boldsymbol{\delta}_{1,2} = (\pm a, 0), \boldsymbol{\delta}_{3,4} = (0, \pm a)$ and results in :

$$\hbar\omega = g\mu_B H_{\text{Zeeman}} + KS_z + JS_z\,(4 - 2\cos(k_x a) - 2\cos(k_y a))$$

Acknowledgements

First, I would like to thank my supervisors Dr. Phivos Mavropoulos and Dr. Samir Lounis for their support and discussions on the topic of this thesis. You always had an open door for me.

Thanks to Prof. Dr. Stefan Blügel for giving me the opportunity to write this thesis in his group at the Forschungszentrum Jülich and for providing an excellent working environment for solid state research.

Last but not least, I would like to thank Andreas Dolfen and Alex Thieß for proofreading my thesis and my girlfriend Kristina Kurth for supporting me during the time.

Bibliography

[1] S. Sun, C.B. Murray, D. Weller, L. Folks, and A. Moser. Monodisperse fept nanoparticles and ferromagnetic FePt nanocrystal superlattics. *Science*, 287(17):1989–1992, 2000.

[2] C. Desvaux, C. Amies, P. Fejes, P. Renaud, M. Respaud, P. Lecante, E. Snoeck, and B. Chaudret. Multimillimetre-large superlattices of air-stable iron-cobalt nanoparticles. *Nature Mater*, 4:750–753, 2005.

[3] G. Reiss and A. Hütten. Magnetic nanoparticles: Applications beyond data storage. *Nature Mater*, 4:725–726, 2005.

[4] P. Gambardella, A. Dallmeyer, K. Maiti, M. C. Malagoli, W. Eberhardt, K. Kern, and C. Carbone. Ferromagnetism in one-dimensional monatomic metal chains. *Nature*, 416:301–304, 2002.

[5] S. Rusponi, T. Cren, N. Weiss, M. Epple, P. Buluschek, L. Claude, and H. Brune. The remarkable difference between surface and step atoms in the magnetic anisotropy of two-dimensional nanostructures. *Nature Mater*, 2:546–551, 2003.

[6] I. Tudosa, C. Stamm, A. B. Kashuba, F. King, H. C. Siegmann, J. Stöhr, G. Ju, B. Lu, and D. Weller. The ultimate speed of magnetic switching in granular recording media. *Nature*, 428:831, 2004.

[7] Gerrits Th., van den Berg H. A. M., Hohlfeld J., Bär L., and Rasing Th. Ultrafast precessional magnetization reversal by picosecond magnetic field pulse shaping. *Nature*, 418:509–512, 2002.

[8] J. Hellsvik. *Atomistic Spin Dynamics*. PhD thesis, University of Uppsala, 2008.

[9] U. Wilke. *Magnetization Dynamics of One-Dimensional Chains Studied by Kinetic Monte Carlo Simulations*. PhD thesis, University of Hamburg, 2008.

[10] K. A. Fichthorn and W. H. Weinberg. Theoretical foundations of dynamical monte carlo simulations. *J. Chem. Phys.*, 95(2):1090.

[11] N. D. Mermin and H. Wagner. Theoretical foundations of dynamical monte carlo simulations. *Phys. Rev. Lett.*, 17,:1133–1136, 1966.

[12] H. Ibach and H. Lüth. *Festkörperphysik*. Springer, 2002.

[13] S. Blügel, T. Brückel, and C. M. Schneider (eds). *Magnetism goes Nano. Forschungszentrum Juelich, Reihe Matter and Materials*, volume 26 p. C2.9-C2.11. 2005.

[14] N. W. Ashcroft and N. D. Mermin. *Festkörperphysik*. Oldenbourg.

[15] `http://www.fz-juelich.de/iff/staff/Bluegel_S/Lectures/mca.pdf`.

[16] S. Blügel, T. Brückel, and C. M. Schneider (eds). *Magnetism goes Nano. Forschungszentrum Juelich, Reihe Matter and Materials*, volume 26 p. C2.7. 2005.

[17] S. Blügel, T. Brückel, and C. M. Schneider (eds). *Magnetism goes Nano. Forschungszentrum Juelich, Reihe Matter and Materials*, volume 26 p. C2.8. 2005.

[18] V. P. Antropov, S. V. Tretyakov, and B. N. Harmon. Spin dynamics in magnets: Quantum effects and numerical simulations. *J. Appl. Phys.*, 81(3961), 1997.

[19] C. Kittel. *Introduction to solid state physics*. 2005.

[20] A. Bulgac and D. Kusnezov. Classical limit for lie algebras. *Annals of Physics*, 1990.

[21] L. Landau and E. Lifschitz. On the theory of magnetic permeability in ferromagnetic bodies. *Physik. Z. Sowjetunion*, 8:153, 1935.

[22] T. L. Gilbert. *Phys. Rev.*, 100:1243, 1955.

[23] B. E. Argyle, W. Jantz, and J. C. Slonczewski. Wall oscillations of domain lattices in underdamped garnet films. *J. Appl. Phys.*, 54(3370), 1983.

[24] N. Inaba, Y. Uesaka, A. Nakamura, M. Futamoto, Y. Sugita, and S. Narishige. Damping constants of Co-Cr-Ta and Co-Cr-Pt thin films. *IEEE Transactions On Magnetics*, 33:2989–2991, 1997.

[25] A. V. Savin, G. P. Tsironis, and X. Zotos. Thermal conductivity of a classical one-dimensional spin-phonon system. *Phys. Rev. B*, 75, 2007.

[26] A. V. Savin, G. P. Tsironis, and X. Zotos. Thermal conductivity of a classical one-dimensional heisenberg spin model. *Phys. Rev. B*, 72, 2005.

[27] `www.physik.uni-augsburg.de/annalen/history/papers/1905_17_549-560.pdf`.

[28] W. Horsthemke and R. Lefever. *Noise-Induced Transitions*. Springer, 1984.

[29] C. W. Gardiner. *Handbook of stochastic methods*. Springer-Verlag, 1985.

[30] P.E. Kloeden and E. Platen. Numerical solution of stochastic differential equations. 2002.

[31] P.E. Kloeden and E. Platen. Relations between multiple ito and stratonovich integrals. *Stochastic Analysis and Applications*, 9(3):311–321, 1991.

[32] R. Pulch. Stochastische differentialgleichungen: Theorie und numerik. *lecture notes*, 2006/07.

[33] E. Kloeden. Numerik stochastischer differentialgleichungen. *lecture notes*, 2007.

[34] H. K. Leung. Stochastic aspects of nonequilibrium phase transitions. *Phys. Rev. A*, 41(4):1862–1866, 1990.

[35] G. N. Milstein and M. V. Tretyakov. Numerical methods in the weak sense for stochastic differential equations with small noise. *SIAM J. Numer. Anal.*, 34(2142):2142–2167, 1997.

[36] C. Schröder. *Nummerische Simulation zur Thermodynamik magnetischer Strukturen mittels deterministischer und stochastischer Wärmebadankopplung*. PhD thesis, University of Osnabrück, 1998.

[37] G. Brown, M. A. Novotny, and Per Arne Rikvold. Transition state in magnetization reversal. *Journal of Applied Physics*, 93(10):6817–6819, 2003.

[38] O. Pietzsch, S. Okatov, A. Kubetzka, M. Bode, S. Heinze, A. Lichtenstein, and R. Wiesendanger. Spin-resolved electronic structure of nanoscale cobalt islands on cu(111). *Phys. Rev. Lett.*, 96(237203).

[39] O. Pietzsch, A. Kubetzka, M. Bode, and R. Wiesendanger. Spin-polarized scanning tunneling spectroscopy of nanoscale cobalt islands on Cu(111). *Phys. Rev. Lett.*, 92(5).

[40] J. Barnes and P. Hut. A hierarchical $\mathcal{O}(n\log n)$ force-calculation algorithm. *Nature*, 324(4), 1986.

[41] L. Greengard and V. Rokhlin. A fast algorithm for particle simulations. *J. Comp. Phys.*, 73(325), 1987.

[42] L. Greengard. *The Rapid Evaluation of Potential Fields in Particle Systems*. MIT, Cambridge, 1987.

[43] C. A. White and M. Head-Gordon. Derivation and efficient implementation of the fast multipole method. *J. Chem. Phys.*, 101(6593), 1994.

[44] G. Brown, T. C. Schulthess, D. M. Apalkov, and P. B. Visscher. Flexible fast multipole method for magnetic simulations. *IEEE Transactions on Magnetics*.

[45] T. Cormen, R. Rivest T. Leiserson, and C. Stein. *Introduction to Algorithms*. MIT Press.

[46] T. Fliessbach. *Elektrodynamik*. Spektrum, 2004.

[47] http://bama.ua.edu/~visscher/mumag/cart.pdf.

[48] H.B. Callen and B. Herbert. Green function theory of ferromagnetism. *Phys. Rev.*, 130(3):890–898, 1963.

[49] F. Bloch. Zur Theorie des Ferromagnetismus. *Z. Physik*, 61, 1930.

[50] S. Blügel, T. Brückel, and C. M. Schneider (eds). *Magnetism goes Nano. Forschungszentrum Juelich, Reihe Matter and Materials*, volume 26 p. A3.13. 2005.

[51] N. D. Mermin and H. Wagner. Absence of ferromagnetism or antiferromagnetism in one- or two-dimensional isotropic heisenberg models. *Phys. Rev. Lett.*, 17(22):1133–1136, 1966.

[52] N. D. Mermin. Crystalline order in two dimensions. *Phys. Rev.*, 176(1):250–254, 1968.

[53] G Reiss and A Hütten. Magnetic nanoparticles: Applications beyond data storage. *Nature Materials*, 4, 2005.

[54] M. Tanase, J.-G. Zhu, C. Liu, N. Shukla, T.J. Klemmer, D. Weller, and D.E. Laughlin. Structure optimization of fept nanoparticles of various sizes for magnetic data storage. *Metallurgical and Materials Transactions A*, 38(4):798–810, 2007.

[55] D. A. e. a. Thompson. *IBM J. Res. Dev.*, 102(1413), 1956.

[56] D. e. a. Weller. *IEEE Trans.Magn.*, 35(4423), 1999.

[57] N. Knorr, , H. Brune, M. Epple, A. Hirstein, M. A. Schneider, and K. Kern. Long-range adsorbate interactions mediated by a two-dimensional electron gas. *Phys. Rev. B*, 65(11):115420, 2002.

[58] S. Tait, Y. Wang, G. Costantini, N. Lin, A. Baraldi, F. Esch, L. Petaccia, S. Lizzit, and K. Kern. Metal-organic coordination interactions in Fe-terephthalic acid networks on Cu (100). *J. Am. Chem. Soc.*, 130(6):2108–2113, 2008.

[59] V. S. Stepanyuk, L. Niebergall, R. C. Longo, W. Hergert, and P. Bruno. Magnetic nanostructures stabilized by surface-state electrons. *Phys. Rev. B*, 70(7).

[60] L. Néel. Theorie du trainage magnetique des ferromagnetiques en grains fins avec applications aux terres cuites. *Ann. Geophys.*, 5:99–136, 1949.

[61] http://www.coyotegulch.com/products/libcoyotl/twisted_road/.

[62] http://www.math.sci.hiroshima-u.ac.jp/~m-mat/MT/emt.html.

List of Figures

2.1 Schematic deformation of orbitals by different spin states 13
2.2 Motion of a spin around an effective magnetic field 17
2.3 Motion of a spin around a magnetic field including a damping term . . . 19
2.4 Brownian Motion of a particle in a liquid 21
2.5 Motion of a spin around an effective field including a heat bath 23
2.6 Example paths of a Wiener process 25
2.7 Distribution for a parallel code by using domain decomposition 29
2.8 Efficiency curve 31
2.9 Trajectory of single spins in a magnetic field including a heat bath 32
2.10 Histogram of the energy distribution of a single spin 33
2.11 Length of a single spin during a simulation run 34

3.1 Spin-wave mode of an infinitely long chain 35
3.2 Dispersion of an infinitely long chain 37
3.3 Spin-resolved STM picture of Co nano-islands 41
3.4 Degenerated spin-modes of a chain 41
3.5 Spinmodes of a one dimensional spin chain 42
3.6 Spin-modes of a nano-disc 44
3.7 Spin-modes of a triangular shaped nanostructure 45
3.8 Density of states for a chain with 20 atoms 47
3.9 Density of states for a two dimensional infinite cubic lattice 47

4.1 Separation of space visualized using a tree structure 53
4.2 Visualization of an different interaction lists 55
4.3 Output sample picture of a two dimensional interaction list 55
4.4 Output sample picture of a three dimensional interaction list 56
4.5 Key labeling of a tree refinement 57
4.6 Procedure for the generation of particle keys 58
4.7 Key ordering path 59
4.8 Parent and Child finding using bitshift operations 59
4.9 Distribution of particles around its center of charge 61
4.10 Merging multipole moments to the parent cell 63
4.11 Accuracy of the multipole calculation 66
4.12 Timing of the multipole calculation 66

5.1 $10 \times 10 \times 10$ cubic lattice of atomic magnetic moments 67
5.2 Magnetization in time of a cubic lattice 68

5.3 Magnetization in time depending on the damping parameter 69
5.4 Magnetization curve of a bulk system with a cubic lattice 71
5.5 One dimensional chain of atomic moments on a surface. 71
5.6 STM image of Cobalt chains which align on the step edges 72
5.7 Magnetization of spin chains with different length 73
5.8 Magnetization of spin chains with different length 74
5.9 Demagnetization time of a 5000 spins chain 74
5.10 Demagnetization time of a spin chain 75
5.11 Formation of a FeCo nano-particle on a superlattice 76
5.12 Nano-cluster of atomic moments on a surface. 76
5.13 Projection of the magnetization of a 4×4 particle 77
5.14 Magnetization switching of a 4×4 cluster with low anisotropy 78
5.15 Magnetization switching of a 4×4 cluster during a switching event . . . 78
5.16 Snapshots of the spin state in a high anisotropy model 81
5.17 Snapshots of the spin state in a low anisotropy model 82
5.18 Histogram of the switching time 84
5.19 Switching times of a 4×4 particle 85
5.20 Determination of parameters by the Arrhenius Néel law 85

A.1 Speedup predicted by Amdahl law 91
A.2 Formation of a domain wall 92

Selbständigkeitserklärung

Hiermit versichere ich, die vorliegende Arbeit selbständig und nur unter Zuhilfenahme der angegebenen Quellen und Hilfsmittel angefertigt zu haben.

David Bauer
Jülich, den 28. Oktober 2006