

MASTER THESIS IN PHYSICS

ATOMISTIC SPIN DYNAMICS WITH
REAL TIME CONTROL OF
SIMULATION PARAMETERS AND
VISUALISATION

by

DANIEL CHRISTOPH BEPPO SCHÜRHOFF

born in

WUPPERTAL

Submitted to the

FAKULTÄT FÜR MATHEMATIK, NATURWISSENSCHAFTEN UND INFORMATIK
of the Rheinisch-Westfälische Technische Hochschule Aachen

in

November 2016

written at the

Institute for Advanced Simulation (IAS)
Forschungszentrum Jülich

Daniel Christoph Beppo Schürhoff: *Master Thesis in Physics*,
Atomistic spin dynamics with real time control of simulation parameters and
visualisation,
© November 2016

SUPERVISORS:

Stefan Blügel
Nikolai Kiselev

LOCATION:

Forschungszentrum Jülich

TIME FRAME:

Nov. 2015 to Nov. 2016

ACKNOWLEDGEMENTS

I would like to express my gratitude to Prof. Dr. Stefan Blügel for giving me the chance of finishing my Master with this thesis at the PGI/IAS-1 in the Forschungszentrum Jülich.

Furthermore I want to thank Dr. Nikolai Kiselev for his supervision and constructive criticism during conceptualisation as well as execution of this work. His code was the starting point for this project and his ideas and comments were particularly helpful in finding bugs and conceptual errors.

Special thanks go to my colleague Gideon Müller who never ceases to help. Ranging from introducing me to the project, over code reviews and debugging sessions to discussions over the physical concepts, \LaTeX , TikZ or PGF, his encouragement and support was significant and is deeply appreciated.

Finally I want to thank my family and friends, who supported and believed in me throughout my studies and this work. Your friendship, constant encouragement and love always brightens my day. I am lucky to have you in my life!

CONTENTS

INTRODUCTION	1
1 ENERGY TERMS OF THE EXTENDED HEISENBERG MODEL	9
1.1 Interaction with an externally applied field	9
1.2 Uniaxial magneto-crystalline anisotropy	10
1.3 Cubic magneto-crystalline anisotropy	10
1.4 Exchange interaction	10
1.5 Dzyaloshinskii-Moriya interaction	11
1.6 Dipole-dipole interaction of magnetic spins	13
1.7 Higher order exchange terms	13
1.8 Effective field	14
2 ATOMISTIC SPIN DYNAMICS	15
2.1 Micromagnetic model vs atomistic Heisenberg model	15
2.2 Landau-Lifshitz-Gilbert equation of spin motion	16
2.2.1 Temperature dependent extension	18
2.3 Spin transfer torque	20
2.3.1 Spin polarisation of electric current in ferromagnets	20
2.3.2 Effects of spin transfer torque	22
2.3.3 Spin dynamics simulation for spin transfer torque	23
2.4 Semi implicit integration of the Landau-Lifshitz-Gilbert equation	25
2.4.1 SIB with perpendicular Spin Transfer Torque	27
3 SIMULATION FRAMEWORK SPIRIT	29
3.1 Live control of simulation parameters	32
3.2 Visualisation	35
3.3 Python scripting of simulations	38
3.4 Current development and vision	39
4 NUMERICAL EXPERIMENTS	41
4.1 Time test	41
4.2 Example of domain wall motion	42
4.2.1 Domain wall motion induced by an applied magnetic field	43
4.2.2 Motion induced by a spin polarised current	46
4.3 Skyrmion motion induced by a spin polarised current	48

4.4 Skyrmions on a track-like sample	52
5 CONCLUSION	55
BIBLIOGRAPHY	59

LIST OF FIGURES

Figure I.1	Conventional hard disk drive (HDD)	2
Figure I.2	Schematic of domain wall racetrack memory	3
Figure I.3	Magnetic skyrmions of Neel and Bloch type	4
Figure I.4	Schematic of skyrmion racetrack memory	5
Figure 1.1	Schematic of bulk DMI in a cubic lattice	11
Figure 1.2	Schematic of surface induced DMI in a cubic lattice .	12
Figure 2.1	Lamor spin precession	17
Figure 2.2	Lamor precession with damping	18
Figure 2.3	Stochastic damped precession in LLG	19
Figure 2.4	Schematic band model plot for majority and minority electrons in a ferromagnet	21
Figure 2.5	STT in LGG equation of motion	22
Figure 3.1	Call structure of iterations in the core library	31
Figure 3.2	Spirit's graphical user interface (GUI) developed with QT	33
Figure 3.3	Hamiltonian and LLG simulation settings	34
Figure 3.4	Configuration and visualisation setting	35
Figure 3.5	Visualisation with Red/Blue colormap and z -filter .	36
Figure 3.6	Visualisation with surface interpolation	36
Figure 3.7	3D skyrmion with isosurface visualisation	37
Figure 3.8	Hopfion in isosurface visualisation	38
Figure 4.1	Simulated Lamor precession of a single spin	42
Figure 4.2	Initial state and relaxed Neel type domain wall	43
Figure 4.3	Precessing steady motion of the domain wall in- duced by an external magnetic field	44
Figure 4.4	Domain wall motion in an external magnetic field .	45
Figure 4.5	Precessing movement of the domain wall pushed by a spin polarised current	46
Figure 4.6	Domain wall motion induced by a spin polarised current	47
Figure 4.7	Skyrmion initialisation	50
Figure 4.8	Skyrmion motion induced by spin polarised current .	51
Figure 4.9	Skyrmion motion in respect to spin transfer torque α_s	52
Figure 4.10	Skyrmion on a track-like sample with open boundaries	53

LIST OF TABLES

Table 4.1	LLG Parameters for domain wall simulation	43
Table 4.2	LLG Parameters for skyrmion simulation	49

ACRONYMS

API	application programming interface
ASD	atomistic spin dynamics
DDI	dipole-dipole interaction
DMI	Dzyaloshinskii-Moriya interaction
GNEB	geodesic nudged elastic band
GUI	graphical user interface
HDD	hard disk drive
LLG	Landau-Lifshitz-Gilbert
LLGS	Landau-Lifshitz-Gilbert-Slonczewski
MMF	minimum mode following
SIB	semi-implicit method B
SPC	spin polarised current
Spirit	simulation framework Spirit
STT	spin transfer torque

INTRODUCTION

Since the industrial revolution, the advent of computer technology was indubitably one of the most radical changes in modern history. Within laptops, smartphones, integrated control circuits in washing machines, microwaves and other devices, the Internet and countless other applications: computers and computing technology are nearly everywhere in our modern day life. One central component of every computing system is memory. The need for saving data is universal and has different characteristic demands like storage capacity, read and write speeds, latency, energy consumption, physical size and persistence time.

One prominent form of memory of persistent bulk memory is magnetic storage as found in a hard disk drive ([HDD](#)) or magnetic tapes. Being easily scalable, having huge storage capacities and a long lifetime, magnetic memory is especially suited for large amounts of data that do not have taxing requirements on access speed. Physically these mediums consist of a material with a high magneto-crystalline anisotropy. While the ground state is ferromagnetic, orientating the magnetic moments into the opposite direction is a local energetic minimum as well. Being protected by the strong uniaxial anisotropy and respective energy barriers, domains of both spin orientations (ground state and opposite direction) are extremely stable and have a very long lifetime even at room temperature. Utilising this, the memory devices employ these domains as single bits. A read/write head traverses the medium and writes information by applying a strong magnetic field, reorientating the current domain. Read-out of information is facilitated by giant magneto resistance effect, resulting in a resistance changing measurably depending on the domain orientation [\[1\]](#)[\[2\]](#).

This approach has the advantage of easy scalability and relative simplicity. The surface area of the magnetic medium divided by the domain size is proportional to storage capacity, this can easily be expanded on tapes or multiple rotating coated glass disks as in [HDDs](#) and shown in [figure I.1](#). While facilitating fast realisation into commercial products, this approach also has significant drawbacks. Firstly the information is saved inherently two dimensional and while multiple layers can be realised, as is common in [HDDs](#), this still principally decreases or limits storage density in comparison to a three dimensional approach. Furthermore either the head or the storage medium has to be mechanically moved to facilitate addressing

every domain, which in turn, has several limitations attached to it. Adding significant latency to accessing distant sectors, wear of mechanical parts or even accidental high-speed contact of the head with the medium are all substantial drawbacks. Furthermore, miniaturisation of nearly all electronic devices progresses further and therefore the demand of devices with high storage density and little power consumption continues to increase in importance. These two key requirements can not be fulfilled by current **HDDs**. As can be seen in [figure I.1](#) only a very small fraction of the device volume is actual data storage. While this can be increased incrementally

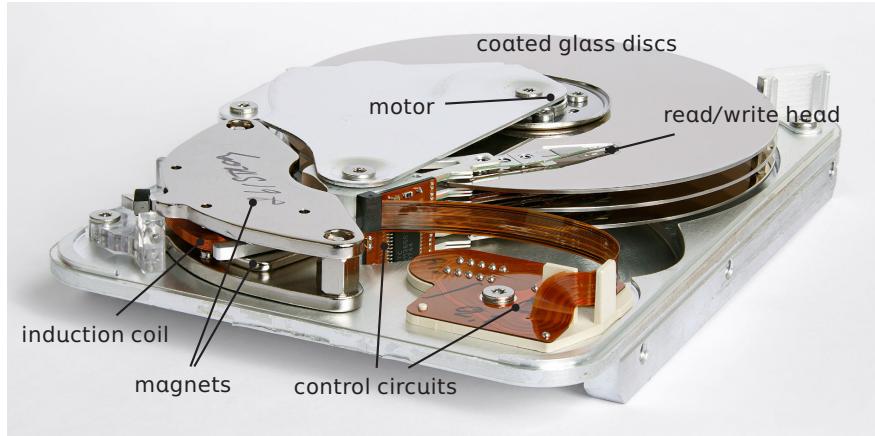


FIGURE I.1 – Conventional hard disk drive (HDD)

Seagate ST33232A HDD with 6 magnetic plates on 3 coated glass discs. The motor spins the discs and the read/write head is moved in/outward by inducing a current into the induction coil surrounded by magnets. As is apparent from the picture the actual data storage is only a fraction of the physical space, resulting in a very low storage density. Original image from [\[3\]](#)

with thinner discs and heads, the general layout is fixed, which severely limits the storage density, since a lot of the physical space is occupied by non-information storing parts like motors, coils and magnets, needed to move the head into different positions. Furthermore the energy consumption can only be reduced marginally due to the fact that the need for rotation of the discs and movement of the head can not be eliminated. This results in an overwhelming amount of power dedicated to operating the physically movable parts, while the actual reading/writing processes only use a tiny fraction of the total power consumption.

The need for alternative solutions for persistent high capacity storage devices is apparent and therefore different technologies have been proposed and researched, including flash memory as utilised in USB sticks and solid state drives, magnetic random access memory, bubble domain memory and racetrack memory. Bubble domain memory was conceptualised as a series of circular magnetic domains, being nucleated and moved on a stationary

chip [4]. While having no movable parts, moving domains with magnetic fields proved to be impractical, due to the high required fields and therefore joule heating problems of coils. With racetrack memory, a similar idea was later conceptualised by Parkin [5][6, 7] in 2004 and offers theoretically unrivalled storage densities and access speeds. Qualitatively it is similar to the classical concept of magnetic storage as described above. However, in this case nanowires are employed to hold the domains which

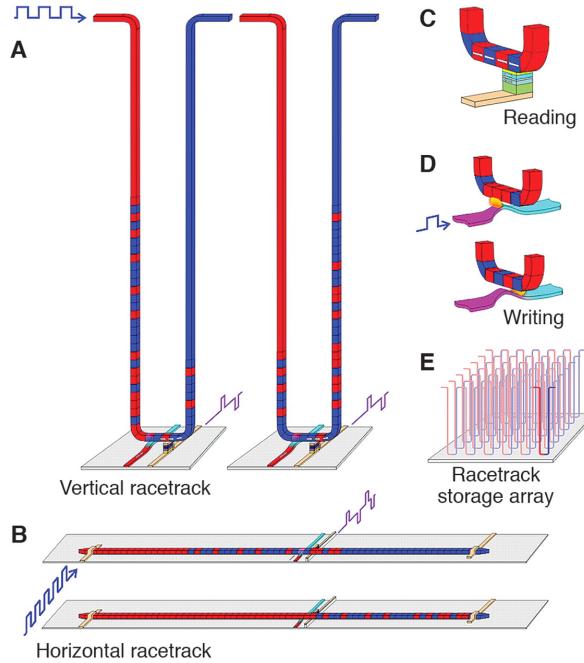


FIGURE I.2 – Schematic of domain wall racetrack memory

In racetrack memory, magnetic domains in nanowires are utilised to store information. The domains can be discretely moved forwards or backwards along the wire by applying a current and therefore read/written by the stationary head, similar to classical **HDDs**, but without moving parts. Image taken from [6]

are then moved along the wires by a current. In contrast to manipulation with a magnetic field, this current can move all domains while passing through the wire. The read and write units are identical to **HDDs**, but since being stationary, effectively eliminating the disadvantages described above. A schematic of this domain wall memory is depicted in [figure I.2](#). In the course of time this idea was refined three times [7] and a proof of concept was demonstrated by [Hayashi et al.](#) [8] in 2008. Despite all the described advantages, research and proof of concept, up to this day no functional prototype has been introduced.

In practice impurities in the described nanowires pose a significant problem. The moving domain walls can be pinned by these impurities, requiring large amounts of current to dislodge and therefore extensively reducing

the domain wall propagation speed by orders of magnitude. This could be remedied by employing higher drive currents, but Joule heating of the nanowires and Walker breakdown of the domain wall motion [9, 10, 11] impose considerable limits to this solution. One of the approaches of circumventing this problem is to employ skyrmions or other non-trivial chiral magnetic structures instead of domains as bits.

Magnetic skyrmions are vortex-like structures as depicted in [figure I.3](#), which may appear in chiral magnetic materials. They correspond to lo-

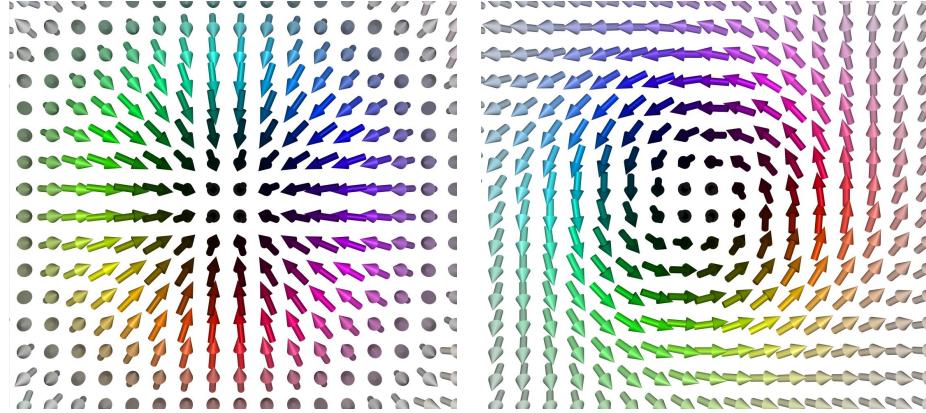


FIGURE I.3 – Magnetic skyrmions of Neel and Bloch type

In the Neel type skyrmion (left) the spins reorient within the radial plane. This type of skyrmion was predicted in magnetic crystals [12, 13] with a C_{nv} symmetry and recently found in GaV_4S_8 [14]. In the Bloch type skyrmion (right) the reorientation happens perpendicular to the radial plane. This skyrmion type was amongst others observed in chiral magnets such as $\text{Fe}_{1-x}\text{Co}_x\text{Si}$ [15], $\text{Mn}_{1-x}\text{Fe}_x\text{Ge}$ [16], Cu_2OSeO_3 [17], FeGe [18], MuSi [19]. Pictures taken with [Spirit](#).

cal energy minima protected by their topology related energy barrier. In practice they may appear as isolated objects in a metastable state or as the ground state of a system in which case they condense into skyrmion lattices. For a certain range of system parameters, in particular exchange stiffness and strength of Dzyaloshinskii-Moriya interaction (**DMI**), these skyrmions can be extraordinarily stable, while retaining a very small size down to just a few nanometres.

This is a major advantage of skyrmions over domain walls concerning race-track memory or information storage in general, since domains can not easily be reduced to less than $30 - 40\text{nm}$ [20]. Skyrmions are therefore able to represent discrete bits of information on a considerably smaller volume. Applying the racetrack concept to skyrmions as conceptualised by [Fert et al.](#) [20] is depicted in [figure I.4](#) and an active topic of research [21, 22, 23].

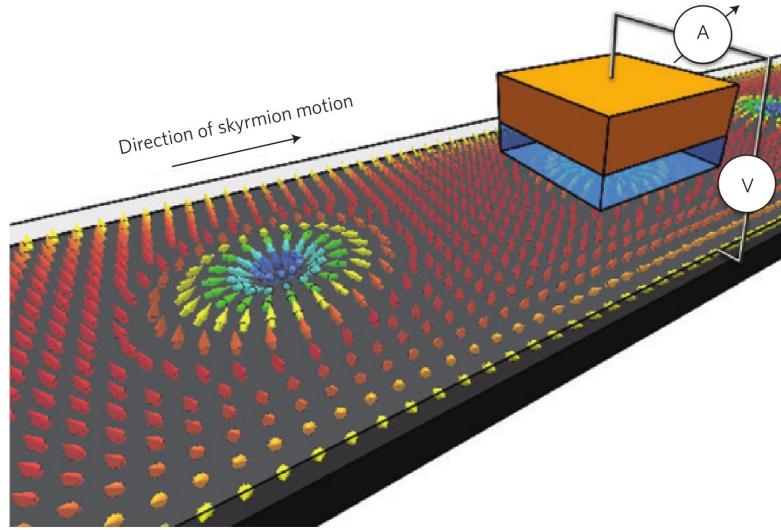


FIGURE I.4 – Schematic of skyrmion racetrack memory

In this version of racetrack memory, skyrmions in nanostripes are utilised to store information. The skyrmions can be moved forwards or backwards along the stripe by applying e.g. a spin polarised current and that way every bit read by the stationary head depicted in orange/blue here. Image taken from [23]

Using skyrmions instead of domain walls as bits significantly reduces the minimum physical space needed to encode a data bit, which in turn increases write, read and access times assuming the same propagation speed per current density. Due to the smaller size of skyrmions and the fact that they are also constrained in a second dimension and do not simply span the whole nanowire, they are also significantly less likely to be pinned by impurities. Because of their small size the chance of encountering an impurity is significantly lower. Additionally, given well chosen parameters, it is in principal possible for the skyrmion to be repelled by the impurity and move around it, literally circumnavigating the problem.

In summary, skyrmions, due to their small size and good stability, are promising candidates for racetrack memory and the ever growing demand of small, low powered data storage. While still being far from adaptation in daily devices, the fundamental research is well underway.

Well explored is e.g. the thermodynamic stability as done in [12, 24, 25, 26, 27] in mainly the micromagnetic model which is outlined in section 2.1. Transport properties are a less explored facet of research investigating skyrmions. Developing theoretical models describing the impact of their topological characteristics and characterising the skyrmion hall effect was amongst others done by [Yin et al.](#) [28] and [Hamamoto et al.](#) [29][30, 31] and recently confirmed experimentally by [Jiang et al.](#)

[32] and [Litzius et al.](#) [33].

The topic of skyrmion dynamics, especially the interactions with spin polarised currents, is another important and actively developing direction of research [34, 35, 36, 37, 38]. Addressing important questions about the behaviour of skyrmions in varying fields and currents and trying to reliably manipulate skyrmions with those tools is important knowledge that ultimately decides the feasibility of application in skyrmion data storage.

This thesis will focus on the development of the comprehensive simulation framework [Spirit](#) [39]. In order to create a tool which enables high productivity and rapid investigations of new phenomena, the existing juSpinX [40] is rewritten from scratch and extended by including spin transfer torque. Besides a new python scripting interface, a graphic user interface is added, featuring in situ control of simulation parameters and real time visualisation.

In [chapter 1](#) the approach of modelling chiral magnetic structures is described. The extended Heisenberg model is introduced with energy terms for the common interactions.

After the model is laid out, [chapter 2](#) characterises the employed numerical model. For differentiation the Micromagnetic Model is introduced and contrasted with the atomistic spin dynamics ([ASD](#)) approach, used in this work. The Landau-Lifshitz-Gilbert ([LLG](#)) equation is presented as an approach to describe the physical dynamics of spin systems. The equation is then extended to include spin transfer torque caused by spin polarised currents into the simulation. A few approximations are discussed to simulate the effect of spin transfer torque onto magnetic moments. Subsequently the widely used semi-implicit method B ([SIB](#))[41], a specialised solver for the [LLG](#) equation, is explained and extended to include the effects of spin transfer torque onto the system.

In [chapter 3](#) the simulation framework [Spirit](#) is introduced by explaining some of the design principles, employed in writing this code. Subsequently the live control, visualisation and python scripting interface are described. The section finishes by giving an outlook on features in current development and the direction, in which [Spirit](#) will move in the future.

In [chapter 4](#) the code is time calibrated with help of the Lamor precession of a single spin. Furthermore easy simulations are done to demonstrate and test the correct behaviour of the simulation framework with non-trivial examples. In order to do that, domain walls are moved with both: magnetic fields and spin polarised currents. After this simpler test case, skyrmion motion under spin polarised current is investigated. Moreover the interaction between skyrmions and open boundaries is observed qualitatively.

The [conclusion](#) summarises the thesis, discusses the employed approximation of the spin polarised current and summarises the simulation capabilities of the code. It then proceeds to provide an outlook, a summary of problems that can be solved by the framework at this point and which further techniques are needed in order to expand the simulation capabilities even further. This is complemented by applications and examples of where those additional features are useful or necessary.

ENERGY TERMS OF THE EXTENDED HEISENBERG MODEL

1

This chapter introduces the extended Heisenberg model, briefly describing the different energy terms and concludes by giving the respective effective fields for all contributions.

While a complete quantum mechanical description of a solid state system would be the most accurate, it is simply impossible to consider systems with multiple atoms efficiently. Therefore a semi classical description is utilised, which tries to represent the quantum mechanical interactions in a classical equation of motion. The basic premise of this model can thus be summarised as treating the solid as an accumulation of independent magnetic moments $\vec{m}_i = \mu_s \cdot \hat{n}_i$, pointing in arbitrary directions \hat{n}_i , with \hat{n}_i being a unit vector $|\hat{n}_i| = 1$ and μ_s being the magnetic moment. Each of these magnetic moments has an energy associated with its interaction with other moments and external fields. These energy contributions together form the Hamiltonian of the extended Heisenberg model.

1.1 INTERACTION WITH AN EXTERNALLY APPLIED FIELD

Magnetic moments in an external field will experience a torque proportional to the angle between the field and the moment and the field strength, as known from classical electrodynamics. Given a homogeneous, time independent external magnetic field \vec{H} the spins will therefore interact with this according to the Zeeman term:

$$\mathcal{H}_{\text{ext}} = -\mu_s H \sum_{i=1}^{N_s} \hat{H} \cdot \hat{n}_i , \quad 1.1$$

with the number of spins N_s , the external magnetic field with respective field strength H and unit vector \hat{H} and the magnetic moment μ_s of a single spin. This term generally favours parallel alignment of the spins along the

external magnetic field, influencing the system towards a ferromagnetic state, with increasing field H .

1.2 UNIAXIAL MAGNETO-CRYSTALLINE ANISOTROPY

Magneto-crystalline anisotropy is the property of a crystal lattice to favour or impede certain axis of magnetisation. Spin orbit coupling is the primary cause of this interaction. The electron motion around the nucleus results in an interaction between the electron spin and the magnetic field caused by its motion, favouring certain spin orientations. Coupled with the structure and symmetry of a lattice, this leads to a anisotropy, here described uniaxial in first order:

$$\mathcal{H}_{ani} = -K \sum_{i=1}^{N_s} (\hat{K} \cdot \hat{n}_i)^2, \quad 1.2$$

with respective magnitude K and unit vector \hat{K} . In this convention a positive anisotropy magnitude $K > 0$ describes a preferred axis and a negative $K < 0$ discourages magnetisation along the specified axis \hat{K} .

1.3 CUBIC MAGNETO-CRYSTALLINE ANISOTROPY

In contrast to the uniaxial case, the anisotropy energy density can also have a cubic symmetry. In this case, the contribution to the hamiltonian is given by

$$\mathcal{H}_{ani,cubic} = -K_1 \sum_{i=1}^N (\hat{n}_{i,x}^2 \hat{n}_{i,y}^2 + \hat{n}_{i,y}^2 \hat{n}_{i,z}^2 + \hat{n}_{i,z}^2 \hat{n}_{i,x}^2) - K_2 \sum_{i=1}^N \hat{n}_{i,x}^2 \hat{n}_{i,y}^2 \hat{n}_{i,z}^2, \quad 1.3$$

with $\hat{n}_{i,x}$, $\hat{n}_{i,y}$ and $\hat{n}_{i,z}$ being the x , y and z -component of spin \hat{n}_i , respectively. This term is dependent on K_1 , where a positive constant favours alignment along the 8 diagonals, and K_2 favouring alignment along the 6 coordinate directions. Depending on the concrete material and it's symmetries, both cubic and uniaxial anisotropy may be required simultaneously to get a good approximation of the anisotropy energy contribution.

1.4 EXCHANGE INTERACTION

As two identical particles are localised close to each other, their wavefunctions can overlap, yielding the exchange interaction. This interaction between two magnetic moments is described by the Heisenberg exchange

term:

$$\mathcal{H}_{\text{ex}} = - \sum_{\langle ij \rangle} J_1 \hat{n}_i \cdot \hat{n}_j - \sum_{\langle\langle ij \rangle\rangle} J_2 \hat{n}_i \cdot \hat{n}_j - \cdots - \sum_{\langle\langle\langle ij \rangle\rangle\rangle} J_N \hat{n}_i \cdot \hat{n}_j , \quad 1.4$$

with $\langle ij \rangle$ denoting all nearest neighbour pairs and $\langle\langle ij \rangle\rangle$ all next nearest neighbour pairs with corresponding interaction constants J_1 , J_2 until J_N for the different shells. Assuming a crystal lattice and isotropic exchange interaction, the exchange constants typically only depend on the distance between two spins and vanish either exponentially or according to a power law with increasing distance, depending on the material properties. This term favours a homogeneous spin orientation with slow changes, pushing the system away from sharp transitions with growing interaction constants.

1.5 DZYALOSHINSKII-MORIYA INTERACTION

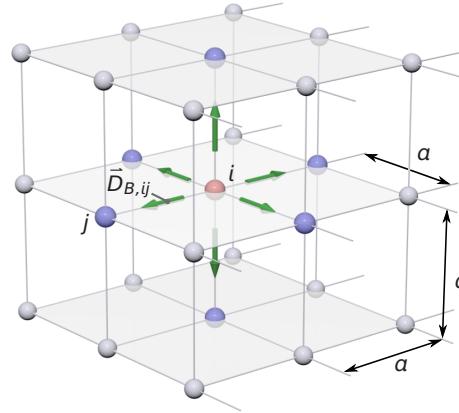


FIGURE 1.1 – Schematic of bulk DMI in a cubic lattice

For every atom i (red) in the bulk of the simple cubic lattice has 6 nearest neighbours (blue). The bulk Dzyaloshinskii-Moriya interaction vectors are visualised in green, pointing along the connection \vec{r}_j to the nearest neighbours. Visualisation adapted from [42]

Originally proposed as a phenomenological model by [Dzyaloshinsky](#) [43], the antisymmetric exchange was also traced back to the spin-orbit coupling by [Moriya](#) [44] in conjunction with a broken inversion symmetry. The contribution of this interaction to the overall Hamiltonian can be denoted as:

$$\mathcal{H}_{\text{DM}} = - \sum_{\langle ij \rangle} \vec{D}_{ij} \cdot (\hat{n}_i \times \hat{n}_j) , \quad 1.5$$

with the [DMI](#) vector \vec{D}_{ij} . Due to the cross product between neighbouring spins, it favours orthogonal spin orientation. The competition of Heisenberg exchange interaction with [DMI](#) can stabilise spin spirals with conserved chirality and handedness or skyrmions, where neighbouring spins

are always slightly turned. The utilised vector can be chosen between "bulk" and "surface" **DMI** in the implementation.

$$\hat{D}_{B,ij} = \hat{r}_{ij} = \frac{\vec{r}_i - \vec{r}_j}{|\vec{r}_i - \vec{r}_j|} \quad \text{and} \quad \hat{D}_{S,ij} = \hat{D}_{B,ij} \times \hat{e}_z . \quad 1.6$$

A visualisation of the bulk and surface DMI in a cubic lattice can be found in figures 1.1 and 1.2 respectively.

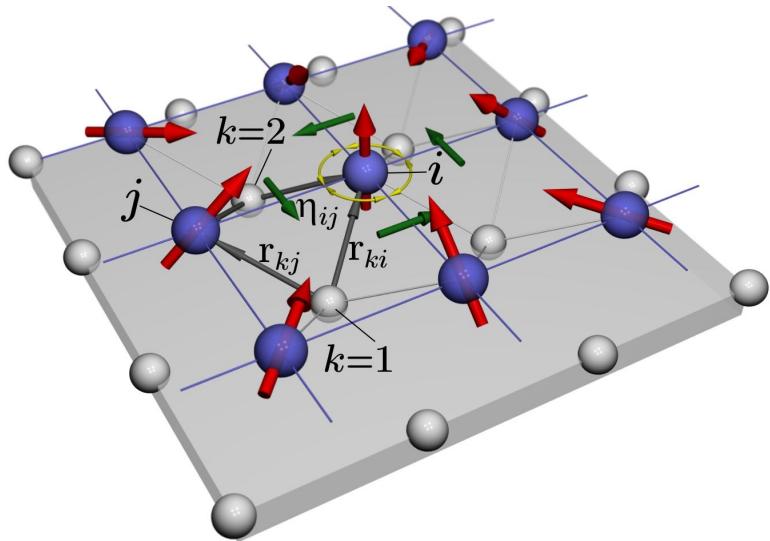


FIGURE 1.2 – Schematic of surface induced DMI in a cubic lattice

Every atom i (blue) in a 2 dimensional surface has 4 nearest neighbours. The surface Dzyaloshinskii-Moriya interaction vectors are visualised in green, pointing perpendicular to the connection \vec{r}_{ij} to the nearest neighbours. This interaction is transmitted via the heavy atoms of the material substrate. Image taken from [45]

From here on isotropic, bulk **DMI** is assumed, resulting in simplification of the **DMI** vector $\vec{D}_{ij} = D\hat{r}_{ij}$ with the **DMI** constant D . Like the exchange interaction, **DMI** couples spins in different shells and the constant D decays either exponentially or according to a power law with increasing distance, depending on the material properties. While generally the interaction can be treated with multiple shells of neighbours, only nearest neighbour **DMI** is assumed in this case. Increasing the **DMI** constant D , favouring of rotating spin orientations is enhanced. Vortex-like structures like skyrmions can be stabilised with this interaction.

1.6 DIPOLE-DIPOLE INTERACTION OF MAGNETIC SPINS

Very intuitive and demonstrated in every basic electromagnetism class with two bar magnets, the magnetic dipole-dipole interaction (DDI) is another way, in which two magnetic moments interact with each other. The Hamiltonian contribution is given by

$$\mathcal{H}_{DD} = -\frac{\mu_0}{4\pi} \sum_{i,j=1, i \neq j}^N \frac{1}{|\vec{r}_{ij}|^3} [3(\vec{m}_i \cdot \hat{r}_{ij})(\vec{m}_j \cdot \hat{r}_{ij}) - \vec{m}_i \cdot \vec{m}_j] , \quad 1.7$$

with \hat{r}_{ij} being the normal vector pointing from spin i to spin j as in [equation 1.6](#), μ_0 the magnetic constant and m_i , m_j the magnetic moments of the spins. Substituting those in with $\vec{m}_x = -\mu_s \mu_B \hat{n}_x$ one gets:

$$\mathcal{H}_{DD} = -\frac{\mu_0 \mu_B^2}{4\pi} \sum_{i,j=1, i \neq j}^N \frac{\mu_{i,s} \mu_{j,s}}{|\vec{r}_{ij}|^3} [3(\hat{n}_i \cdot \hat{r}_{ij})(\hat{n}_j \cdot \hat{r}_{ij}) - \hat{n}_i \cdot \hat{n}_j] , \quad 1.8$$

thus conforming to the notation of all other terms above.

While exchange interaction and [DMI](#) are quantum mechanical effects, decrease either exponentially or according to a power law and are often considered to be short ranged, the [DDI](#) is an electromagnetic interaction, decreasing only proportional to \vec{r}_{ij}^3 . Considered as therefore having a medium to long ranged effect, the [DDI](#) is somewhat problematic in an [ASD](#) simulation: A total calculation of the interaction of every possible spin pair would scale quadratically with the number of spins n for every iteration. To control the trade-off between accuracy of the [DDI](#) and computation time, only spins j within a certain radius $|\vec{r}_{ij}| < r_{DD}$ around spin i are considered for the [DDI](#). The contributions of all others are assumed to be small and therefore neglected. Giving the user control over this process, a parameter 'dd_radius' was introduced to control this radius from the config or during runtime.

The [DDI](#) generally can favour either ferromagnetic or anti-ferromagnetic states depending on the concrete geometry.

1.7 HIGHER ORDER EXCHANGE TERMS

In [section 1.4](#) the first order term of wavefunction overlap was introduced and the effect onto the magnetic moments discussed. Expanding this to

higher order terms, one gets the biquadratic exchange interaction

$$\mathcal{H}_{BQE} = - \sum_{\langle ij \rangle}^N B_{ij} (\hat{n}_i \cdot \hat{n}_j)^2, \quad 1.9$$

with the biquadratic exchange constant B_{ij} and the four-spin interaction

$$\mathcal{H}_{4SE} = - \sum_{\langle ijkl \rangle}^N K_{ijkl} [(\hat{n}_i \cdot \hat{n}_j)(\hat{n}_k \cdot \hat{n}_l) + (\hat{n}_i \cdot \hat{n}_l)(\hat{n}_j \cdot \hat{n}_k) - (\hat{n}_i \cdot \hat{n}_k)(\hat{n}_j \cdot \hat{n}_l)], \quad 1.10$$

with the four-spin exchange constant K_{ijkl} and the quadruplet $\langle ijkl \rangle$ of neighbouring spins. While these higher order terms of the exchange interactions are important in some special cases like metamagnetism in FeRh [46], generally both B_{ij} and K_{ijkl} are small and therefore these interactions are commonly omitted.

Summing equations 1.1, 1.2, 1.3, 1.4, 1.5, 1.8, 1.9 and 1.10 yields the full Hamiltonian of the extended Heisenberg model, which can be used in atomistic spin dynamics simulations, which were developed and used in this work.

1.8 EFFECTIVE FIELD

Being able to calculate the total energy of a given system, with help of the above described Hamiltonian, the next step is to calculate the time dependent effective motion of interacting spins. To achieve that, the effective field has to be calculated by taking the negative energy gradient:

$$\mathcal{H}_{i,eff} = -(\nabla E_{tot})_i = -\frac{\partial \mathcal{H}}{\partial n_i}. \quad 1.11$$

Summing the effective field terms (in that order) for interaction with external field, uniaxial anisotropy, exchange interaction and Dzyaloshinskii-Moriya interaction

$$\mathcal{H}_{i,eff} = -\mu_s \vec{H} - 2\vec{K}(\hat{K} \cdot \hat{n}_i) - \frac{1}{2} J_{ij} \hat{n}_j - \frac{1}{2} \vec{D}_{ij} \times \hat{n}_j \quad 1.12$$

of equations 1.1, 1.2, 1.4 and 1.5.

The total effective field summarises the effects of all interactions onto the considered spin. Together with the methods introduced in the following chapter this field vector is employed to calculate the force acting upon every spin and therefore iterating the system.

This chapter lays out the basic approximations and techniques, employed to solve the extended Heisenberg model of [chapter 1](#). It starts by giving the basic assumptions of the micromagnetic model, the atomistic model is introduced as a more accurate description, especially on smaller scales. The applicability and differences of both approaches are discussed shortly and the atomistic spin dynamic approach explained in more detail. Subsequently the Landau-Lifshitz-Gilbert (LLG) equation is introduced as an approach to simulated the time dependent movement of the magnetic moments. Finally the semi-implicit method B integration scheme is discussed as a specialised solver for the LLG equation.

2.1 MICROMAGNETIC MODEL VS ATOMISTIC HEISENBERG MODEL

The micromagnetic model is a commonly used approximation to investigate magnetic phenomena on a scale between a few nm up to a few microns μm . At the core is the assumption that effects on the size of a few atoms or molecules can be neglected and averaged over by considering a continuous unit vector field \vec{m} instead of the discrete set of atoms and their respective magnetic moments \vec{n}_i . This vector field assumed to behave smoothly and well defined, excepting singularities like Block points. Taking this continuum approximation and applying it to the Hamiltonian, one ends up with

$$\mathcal{H}_{MM} = \int_V \mathcal{F}_{ext} + \mathcal{F}_{ani} + \mathcal{F}_{ex} + \mathcal{F}_{DMI} \, d\vec{r} , \quad 2.1$$

where $\int_V d\vec{n}$ denotes integration over the whole considered space or sample and $f_i(\vec{n})$ are the different energy terms, corresponding to the Hamiltonian terms as described in [chapter 1](#), specifically

$$\mathcal{F}_{ext} = M_s \vec{H} \cdot \vec{m} , \quad \mathcal{F}_{ani} = \mathcal{K} (\hat{K} \vec{m})^2 , \quad 2.2$$

with the applied magnetic field \vec{H} , the magnetisation M_s , the constants of uniaxial anisotropy \mathcal{K} in the micromagnetic model. Furthermore

$$\mathcal{F}_{ex} = \mathcal{A}(\nabla \vec{m})^2, \quad \mathcal{F}_{DMI} = \mathcal{D}(\vec{m} \cdot (\nabla \times \vec{m})) , \quad 2.3$$

with the micromagnetic constants \mathcal{D} of Dzyaloshinskii-Moriya interaction and \mathcal{A} for the Heisenberg exchange, to specify only the most important terms.

While the above introduced functionals are easier to evaluate and simulate due to their smoothness and integrability, they are somewhat imprecise when describing small systems. Particularly for phenomena of the size between a few nm and a few atom sizes, this description is not applicable at all, since the assumption, that the vector field \vec{m} varies slowly and smoothly is violated in that case.

The atomistic spin dynamics approach, utilised in this work, on the other hand, does not rely on the continuum approximation, but simulates every molecule/atom as a single magnetic moment. While being more computationally expensive, small non-trivial magnetic structures like skyrmions can be resolved more accurately and materials with random impurities or other anisotropic interactions can be treated. It is even possible to define the interactions between any two particles, or between a particle and the anisotropy or magnetic field independently, enabling simulation of arbitrarily complicated systems or materials.

2.2 LANDAU-LIFSHITZ-GILBERT EQUATION OF SPIN MOTION

The problem of simulating the single magnetic moments as described in the above section is usually tackled by finding the equation of motion for the magnetic moments. This is done by calculating for every spin the reaction in respect to the current effective field which is evaluated as shown in [section 1.8](#).

The starting point to motivate this semi classical equation of motion is the quantum mechanical equation of motion of an arbitrary operator \mathcal{O} given by the commutator:

$$i\hbar \frac{\partial \mathcal{O}}{\partial t} = [\mathcal{O}, \mathcal{H}] , \quad 2.4$$

one can then introduce the Lie-Poisson bracket as the classic equivalent to

a commutator. For a spin system this is given by:

$$\{F, G\} = \sum_{i=1}^{N_s} \epsilon_{\alpha\beta\gamma} \frac{\partial F}{\partial n_{i,\alpha}} \frac{\partial G}{\partial n_{i,\beta}} n_{i,\gamma} , \quad 2.5$$

with a set of vectors $\{\vec{n}\}$ and the function F, G operating on this set of vectors. Now substituting these functions with $F = n_i^\alpha$ and $G = n_j^\beta$, immediately results in:

$$\{n_i^\alpha, n_j^\beta\} = \epsilon_{\alpha\beta\gamma} \delta_{ij} n_i^\gamma . \quad 2.6$$

To get the equation of motion, this classical commutator equivalent can be utilised in conjunction with the above described Hamiltonian:

$$\{n_i^\alpha, \mathcal{H}\} = \hbar \frac{\partial n_i^\alpha}{\partial t} = \sum_{j=1}^N \epsilon_{\alpha\beta\gamma} \delta_{ij} \frac{\partial \mathcal{H}}{\partial n_j^\beta} n_j^\gamma = \left(\frac{\partial \mathcal{H}}{\partial \hat{n}_i} \times \hat{n}_i \right)^\alpha . \quad 2.7$$

Applicable for all vector components α , one can employ the definition of the effective field in [equation 1.11](#) to receive

$$\hbar \frac{\partial \hat{n}_i}{\partial t} = - \frac{\partial \mathcal{H}}{\partial \hat{n}_i} \times \hat{n}_i = - \hat{n}_i \times \vec{H}_{i,\text{eff}} . \quad 2.8$$

This term describes the so-called Lamor precession of the magnetic moment around the effective field which is depicted in [figure 2.1](#) with a

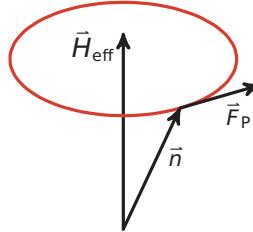


FIGURE 2.1 – Lamor spin precession

The spin \hat{n}_i precesses around the effect field $\vec{H}_{i,\text{eff}}$ acting onto the same spin according to the Lamor precession specified in [equation 2.8](#). Being orthogonal to the effective field $\vec{H}_{i,\text{eff}}$ and the magnetic moments \hat{n}_i conserves the energy and the magnetic moment's strength respectively. Image taken from [42]

force perpendicular to both spin unit vector \hat{n}_i and effective field $\vec{H}_{i,\text{eff}}$ this precession conserves the magnetic moment as well as the spin's energy.

Aiming to relax the system to the minimum energy, [Landau and Lifschitz](#). [47] introduced a damping term in 1935, yielding the Landau-Lifschitz equation

$$\frac{\partial \hat{n}_i}{\partial t} = - \frac{1}{\hbar} (\hat{n}_i \times \vec{H}_{i,\text{eff}}) - \lambda \hat{n}_i \times (\hat{n}_i \times \vec{H}_{i,\text{eff}}) , \quad 2.9$$

with the phenomenological damping parameter λ , which is in this convention assumed to be positive, in order to damp the system in direction of the nearest local energy minimum, represented by the effective field $\vec{H}_{i,\text{eff}}$. This damping term is visualised in [figure 2.2](#) Twenty years later,

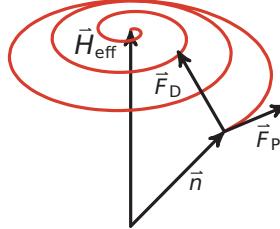


FIGURE 2.2 – Lamor precession with damping

Adding the damping term to the Lamor precession, the magnetic moment \hat{n}_i slowly converges to the effective field $\vec{H}_{i,\text{eff}}$ while exhibiting Lamor precession. This damping relaxes the system towards the nearest local energy minimum while still conserving magnetic moment strength $|\vec{m}_i|$, since both damping and precession forces are orientated perpendicular to it. Image taken from [\[42\]](#)

[Gilbert](#) [\[48\]](#)[\[49\]](#) proposed a similar damping term, yielding the commonly employed [LLG](#) equation

$$\frac{\partial \hat{n}_i}{\partial t} = -\frac{1}{\hbar}(\hat{n}_i \times \vec{H}_{i,\text{eff}}) - \frac{\alpha}{\hbar} \hat{n}_i \times \left(\frac{\partial \hat{n}_i}{\partial t} \right), \quad 2.10$$

with the damping parameter α . The [LLG](#) equation 2.10 can be transformed into [equation 2.9](#) by taking the cross product of [equation 2.10](#) with \hat{n}_i , followed by some simple calculation as shown in [\[50\]](#). While being practically identical for low damping values, the two different formulations only differ significantly for damping values above $\lambda > 0.1$. In that case, the Gilbert damping is generally preferred, since its behaviour in the high damping limit is analogous to classic mechanical damping terms and it is able to model real ferromagnets with higher accuracy with high damping values [\[51\]](#). From here on, the Gilbert version of damping is meant and implemented, whenever damping or damping parameters are referred to.

2.2.1 Temperature dependent extension

So far completely neglecting temperature influences onto the system, one can extend the described model to non-zero temperatures in a few ways. In this work, a Langevin approach is chosen over alternative Monte Carlo methods or a coupling of the spin system to the heat bath of the lattice. Motivated by the intuitive Brownian motion, a fluctuation term \vec{f}_i is added. Assuming directional, spatial and time isotropy for these fluctuations yields

the average of fluctuations

$$\langle f_{i,\alpha} \rangle = 0 \quad 2.11$$

and the correlations

$$\langle f_{i,\alpha}, f_{j,\beta} \rangle = \delta_{ij} \delta_{\alpha\beta} \delta(t - t') \epsilon^2. \quad 2.12$$

Employing the fluctuation-dissipation theorem

$$\epsilon^2 = 2\lambda k_B T, \quad 2.13$$

it is possible to relate these fluctuation correlations to the well known Boltzmann distribution

$$p(E) \propto \exp\left(\frac{-E}{k_B T}\right), \quad 2.14$$

with probability $p(E)$ of encountering a particle at energy E and the temperature T . Knowing that the equilibrium energy distribution of a single particle has to follow [equation 2.14, 2.13](#) can then be used to pick an appropriate fluctuation amplitude ϵ . The effect of the [LLG](#) equation

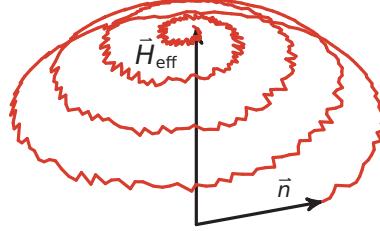


FIGURE 2.3 – Stochastic damped precession in LLG

Adding a stochastical damping term to the [LLG](#) equation, yields a fluctuating, damped Lamor precession (c.f. [figure 2.2](#)). As with all other terms above, the the magnetic moment strength $|\vec{m}|$ is conserved, since the fluctuation term is also applied perpendicular to it. Image taken from [\[42\]](#)

extended by a fluctuation term

$$\frac{\partial \hat{n}_i}{\partial t} = -\frac{1}{\hbar}(\hat{n}_i \times \vec{H}_{i,\text{eff}}) - \frac{\alpha}{\hbar} \hat{n}_i \times \left(\frac{\partial \hat{n}_i}{\partial t}\right) - \hat{n}_i \times \vec{f}_i, \quad 2.15$$

is visualised in [figure 2.3](#). This facilitates the simulation of systems held at finite temperatures by heat baths and completes the discussion of the [LLG](#) equation, it's different terms and their effects onto a single spin.

2.3 SPIN TRANSFER TORQUE

This section starts by explaining the origin of spin polarised currents in ferromagnetic materials and subsequently using this to introduce the phenomenon of spin transfer torque. Finally the utilised approximation is explained by outlining the assumption, applications and restrictions.

2.3.1 *Spin polarisation of electric current in ferromagnets*

In an ordinary current the electron spins are randomly distributed in all directions. Therefore interactions between the electrons spins and e.g. a ferromagnet's particles average out over the large amount of particles and are therefore statistically irrelevant. However, ferromagnets can act as spin polarisers. Following the Stoner model of ferromagnetism, electron bands in a ferromagnet can split up, due to exchange interaction

$$\epsilon_{\uparrow}(k) = \epsilon_0(k) - I \cdot \frac{n_{\uparrow} - n_{\downarrow}}{n} , \quad \epsilon_{\downarrow}(k) = \epsilon_0(k) + I \cdot \frac{n_{\uparrow} - n_{\downarrow}}{n} , \quad 2.16$$

with $\epsilon_{\uparrow}, \epsilon_{\downarrow}$ being the energies of the spin up and down, k the wave vector, $\epsilon_0(k)$ the energy before exchange effects, I the Stoner parameter and $\frac{n_{\uparrow}}{n}, \frac{n_{\downarrow}}{n}$ the spin densities for the up and down electrons respectively. The so-called exchange splitting ΔE_{ex} of the d-band is depicted in [figure 2.4](#) with the density of states $D(E)$ and the Fermi energy level E_F .

When an electron enters the ferromagnet from a nonmagnetic material, the electron enters approximately at the Fermi energy E_F . Being a "spin up" \uparrow (red) electron it can only occupy the $s-p$ band as shown in [figure 2.4](#). The states in the d band are completely occupied, due to the exchange splitting and therefore the only possibility is the $s-p$ band. For an "spin down" \downarrow (blue) electron, however, both the $s-p$ band, as well as the d band are available. Therefore a significant portion of the "spin down" electrons occupy the d band.

Since the $s-d$ band is the overlap integral between s and p band of neighbouring atoms, the incident electron traverses the ferromagnet according to its initial momentum while occupying this band. The d band on the other side is a localised to every atom and therefore prohibits further propagation of the electrons through the material. Those get absorbed by the d band of the corresponding atom, later re-emitted into a random direction and reabsorbed, effectively negating their participation in the currently altogether.

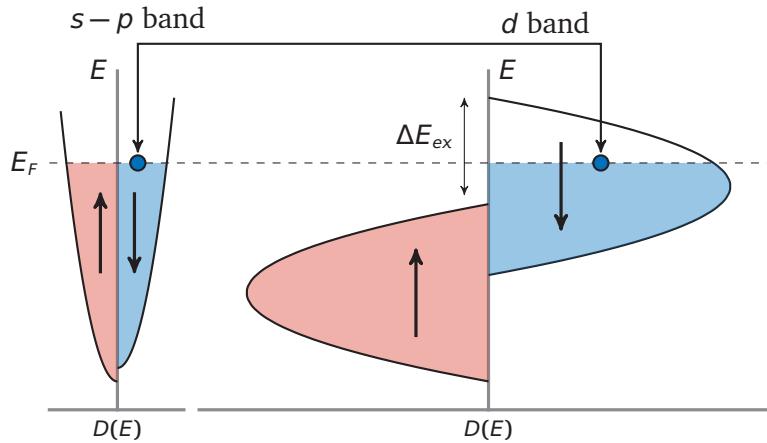


FIGURE 2.4 – Schematic band model plot for majority and minority electrons in a ferromagnet

Only the "spin down" \downarrow (blue) electrons of the $s-p$ band can transition easily into the free states of the d band. The "spin up" \uparrow (red) electrons can not transition into the d band since there are no free states due to the exchange splitting. This results in majority electrons (red) staying in the $s-p$ band and more minority electrons (blue) scattering into the d band.

Summarising, the "spin up" or majority electrons (how they will be called from now) can only join the $s-p$ band and therefore propagate nicely through the ferromagnet. The "spin down" or minority electrons, however, can scatter into the d band at every atom of the ferromagnet, effectively annihilating their contribution to the net current (in an ideal material). This leads to a spin polarisation along the magnetisation direction of a current traversing a ferromagnetic material. For a realistic material, the polarisation is not perfect, but one still ends up with spin dependent conductances, significantly favouring the majority polarisation.

These different conductances of the different polarisation direction of the current, lead to a different scattering behaviour: While the majority electrons are mostly transmitted through the ferromagnetic material, the minority electrons are more often reflected.

In a quantum mechanical picture, an electron with random polarisation can be expressed as the superposition of a majority and a minority spin:

$$P[\uparrow] = \frac{2 + \cos \theta}{2} \quad \text{and} \quad P[\downarrow] = \frac{2 - \cos \theta}{2} , \quad 2.17$$

with θ being the angle between spin and majority spin-state, here taken as spin up $[\uparrow]$.

At the interface, the majority polarised part of the wave function can easily propagate into the ferromagnetic material, while the minority polarised part is mostly reflected and if transmitted, decays exponentially in the

material. As a consequence only the majority part of the spin superposition is easily transmitted and therefore arbitrary incoming conduction electrons are spin polarised by the ferromagnetic material.

2.3.2 Effects of spin transfer torque

In [section 2.3.1](#) we assumed for simplicity, that this process has no effect on the ferromagnet itself. However, considering Newton's third law, one has to take into account the reaction of the above described spin polarisation. The torque needed to rotate the electron's spin, results in a back-torque acting onto the spins of the material itself, the so-called spin transfer torque (STT). First predicted simultaneously by both [Slonczewski](#) [52][53] and [Berger](#) [54][55] in 1996, it was later confirmed experimentally by [Myers et al.](#) [56], [Katine et al.](#) [57] and [Grollier et al.](#) [58] and was added to the [LLG equation 2.10](#) by [Slonczewski](#) [52] in 1996, resulting in the Landau-Lifshitz-Gilbert-Slonczewski (LLGS) equation:

$$\frac{\partial \hat{n}_i}{\partial t} = -\frac{1}{\hbar}(\hat{n}_i \times \vec{H}_{i,eff}) - \frac{\alpha}{\hbar}\hat{n}_i \times \left(\frac{\partial \hat{n}_i}{\partial t}\right) + \tau_{\parallel} \frac{\hat{n}_i \times (\hat{p} \times \hat{n}_i)}{|\hat{p} \times \hat{n}_i|} + \tau_{\perp} \frac{\hat{p} \times \hat{n}_i}{|\hat{p} \times \hat{n}_i|}, \quad 2.18$$

with the driving torques τ_{\parallel} and τ_{\perp} and the polarisation direction \hat{p} of the spin polarised current.

A visualisation of how additional term affects the motion of a single spin in an effective field can be found in [figure 2.5](#). As shown the current can

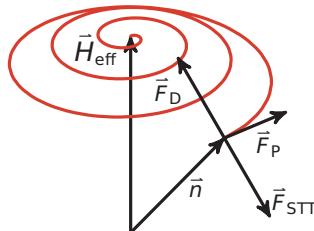


FIGURE 2.5 – STT in LGG equation of motion

Additional to the above described forces of the [LLG](#) equation (c.f. [figure 2.2](#)), here the spin transfer torque is added. The corresponding force term is pointed anti parallel in respect to the damping term. Therefore the system can, depending on the size of those competing terms, either be damped towards the effective field or pushed in the opposite direction by the [STT](#). Since \vec{F}_{STT} is perpendicular to \hat{n} , the length of the magnetic moment $|\vec{m}|$ is conserved.

have two different effects on a single spin:

In the small current limit, the damping term of the material is dominating the interactions. This damping cancels out the spin torque, resulting in a stable system which relaxes to the nearest local minimum with an effec-

tive damping term that is the difference between the original and the **STT**.

However, if the damping is sufficiently small or the **STT** significant enough, the **STT**-term can start dominating [equation 2.18](#) and therefore can excite the system out of an energy minimum and pushing it into different states. The energy gain associated with this term of the **LLGS** equation is therefore not necessarily always suited to relax a system towards minimum energy.

2.3.3 *Spin dynamics simulation for spin transfer torque*

This work simulates the effects and influence of the above described spin transfer torque on non-trivial spin structures like magnetic skyrmion in chiral magnets. As mentioned in the [Introduction](#), so far only magnetic interaction is used in memory devices to read and write information. While also able to move domain walls, external magnetic fields are not suited for the domain wall pushing required in racetrack memory, since they just expand the domains aligned parallel to them and shrink the antiparallel ones. Furthermore application of a current instead of a magnetic field drastically reduces the required energy and therefore significantly improves upon the problem of Joule heating. For this application spin polarised current (**SPC**) and the associated **STT** is valuable.

When applying a spin polarised current onto a series of domains, it is possible to move all those domains simultaneously into the same direction. Being the basic principle of racetrack memory, as described above, this can also be applied to skyrmions. To investigate these interactions further, it is of great importance to add **STT** to the Atomistic Spin Dynamics Simulation.

2.3.3.1 *Complete Spin Transfer Torque treatment*

The intuitive approach to including **STT** into an **ASD** simulation, would be to replace the **LLG** equation of [equation 2.10](#) with the **LLGS** equation of [equation 2.18](#). For that to work, the simulation would be required to track every single electron, the current consists of, with its respective polarisation. While this would then allow for a very detailed and accurate simulation of arbitrary materials and interfaces with a complete treatment of **STT**, it is impractical due to the computational limitation of modern computers. Therefore several approximations can be made to reduce and simplify the otherwise impossible amount of data and calculations.

2.3.3.2 *Calculation via gradient*

One commonly used approximation to implement **STT** into **ASD** simulation with achievable memory and calculation requirements, is the calculation

via gradient. In a bulk material the current direction is specified and the polarisation at the material interface specified. For every spin position \vec{r}_i in the lattice the spin polarisation of the current is then calculated as following.

With the direction of the current, the trail of spins that the current had to pass to get to \vec{r}_i can be calculated. Assuming these spins act as the above described ferromagnet (c.f. [section 2.3.1](#)) polarising the **SPC** in their direction, one can calculate the polarisation direction of the current at a given point \vec{r}_i by taking the gradient of the spin vector field.

While still taxing on the calculation side of the simulation, this approach facilitates the simulation of arbitrary **SPC** with definable accuracy and details. Parameters can be introduced to give fine-grained control over weighting of the current trail according to known material parameters like mean free path and therefore enabling a very detailed and accurate simulation.

2.3.3.3 *Perpendicular to thin layer approximation*

If one is only concerned with reasonably thin layers of materials (nanostripes) and is willing to limit the current flow direction along the axis perpendicular to the stripe, one can simplify the calculations significantly. By assuming the average interaction of a current electron as low within the thin stripe it traverses, the spin of the current's electrons can be assumed to be fixed in their initial polarisation direction.

This approximation is somewhat limited in its generality since one current direction is simulated and for systems with many $z-$ layers, the gradient approximation would be preferred. On the other hand, for thin systems with perpendicular current, it does not need to keep calculating or saving different spin polarisations for every lattice point. This results in an enormously simplified calculation and significant speed advantage in comparison to a calculation via gradient. The concrete quantification for how many layers of a given material this approximation is valid for can be made, employing tools like the mean free path and material constants describing the strength of the spin polarisation by the material. However, since only one-atoms layers are considered by this work, the assumption is indubitably accurate and applicable.

Adding the a Spin torque effect motivated by a spin polarised current to the [LLG equation 2.10](#) as shown by [Li and Zhang \[59\]](#) results in:

$$\frac{\partial \hat{n}_i}{\partial t} = -\frac{1}{\hbar}(\hat{n}_i \times \mathcal{H}_{i,eff}) + \frac{\alpha}{\hbar}\hat{n}_i \times \frac{\partial \hat{n}_i}{\partial t} + \frac{a_j}{\hbar}\hat{n}_i \times (\hat{n}_i \times \hat{S}_p) , \quad 2.19$$

where \hat{S}_p is unit vector describing the polarisation direction of the spin polarised current. While a_j has the units of an effective field, it is proportional and can be related to the current density as shown by [Slonczewski \[52\]](#). Evaluating the cross product in the damping term, this can be phrased in the explicit Gilbert form as shown by [Chureemart et al. \[60\]](#):

$$\begin{aligned} \frac{\partial \hat{n}_i}{\partial t} = & -\frac{\gamma}{(1+\alpha^2)} (\hat{n}_i \times \mathcal{H}_{eff}) - \frac{\gamma\alpha}{(1+\alpha^2)} [\hat{n}_i \times (\hat{n}_i \times \mathcal{H}_{eff})] \\ & - \frac{\gamma\alpha a_j}{(1+\alpha^2)} (\hat{n}_i \times \hat{S}_p) + \frac{\gamma a_j}{(1+\alpha^2)} [\hat{n}_i \times (\hat{n}_i \times \hat{S}_p)] . \end{aligned} \quad 2.20$$

This thin layer approach was implemented into the code as described in [section 2.4.1](#) where the [SIB](#) method is extended to model this modified [LLG](#) equation.

2.4 SEMI IMPLICIT INTEGRATION OF THE LANDAU-LIFSHITZ-GILBERT EQUATION

With the equation of motion layed out in equations [2.10](#) and [2.15](#), the integration method is discussed in this section. Generally there are multiple ways of solving a partial differential equation like the [LLG](#) equation of the previous sections. An analytic solution is the most elegant way: When found, it is applicable universally and provides every desired concrete solution very little further computations. While being theoretically desirable, in complicated N -body problems it is, however, not generally possible to find an explicit closed analytic form. Therefore most complex differential equations like the [LLG](#) are solved numerically.

Starting with the simple Euler method [\[61\]](#) in 1755, a whole set of different numerical integration methods have been devised. The most important characteristics to measure these methods are stability and speed. While it is advantageous to have a small algorithm which needs less calculations for an iteration, the trade-off with stability is important. A more precise or more stable algorithm can utilise bigger time steps τ and therefore need less iterations. Generally there is a distinction between explicit and implicit integration. While explicit integration methods are usually fast and easily implemented, their implicit counterparts are slower, but usually more stable. Since they require solving a system of coupled equations at every iteration step, the implicit methods are often employed for special symmetries or small systems, where these solutions are easily computable and result in a fast and stable iteration.

Of high importance for atomistic spin dynamics with the [LLG](#) equation are especially the Heun method [62][63] and the semi-implicit method B ([SIB](#)) introduced by [Mentink et al.](#) [41] in 2010. The first is an explicit numerical integration algorithm, that can be described as a second order extension of the Euler method. Given a starting point $x(t_0) = x_0$ of a one dimensional function $x(t)$, it iterates the time step τ by performing the simple Euler approximation to get an estimated value

$$\tilde{x}_{k+1} = x_k + \tau \frac{dx_k}{dt} , \quad 2.21$$

with k denoting the number of iteration and then uses this prediction to get the final approximation by again employing the Euler method and averaging both values:

$$x_{k+1} = x_k + \frac{\tau}{2} \left(\frac{dx_k}{dt} + \frac{d\tilde{x}_{k+1}}{dt} \right) . \quad 2.22$$

Extending this to $3N$ dimensions for the [ASD](#) model, one gets a fast explicit iterator, which is easily implementable and commonly used. However it is severely limited concerning its stability and therefore applicable time steps, usually requiring time steps between 0.01 and 1fs [41].

On the other hand, while having high stability, implicit integrations like implicit midpoint are not practical due to the dimensionality of $3N$ and the solution of a $3N$ set of coupled equations is simply too slow. However, still retaining some of the stability advantage, are semi-implicit integration methods. Conserving the length of the individual spins intrinsically, these semi implicit only require the solution of 3 linear coupled equations per spin every iteration. The semi-implicit method B ([SIB](#)) method of [Mentink et al.](#) [41] is a specific semi implicit solver for the [LLG](#) equation. It works with a predictor followed by a correction step. Similar to the Heun method, the predictor is leaned on the Euler method, but here modified to conserve spin length naturally:

$$\tilde{s}_k = \bar{s}_k^i + \tau \frac{\bar{s}_k^i + \tilde{s}_k^i}{2} \times a_i(\bar{s}_k) , \quad 2.23$$

with k denoting the number of iterations as above, \bar{s}_k^i and \tilde{s}_k^i being three dimensional unit vectors, denoting the magnetisation direction of spin i at iteration k and $a_i(\bar{s}_k)$ being the effect of the [LLG](#) equation onto spin i at iteration k , in this case without temperature fluctuations:

$$a_i(\bar{s}_k^i) = -H_{eff}^i(\bar{s}_k) - \alpha \bar{s}_k^i \times H_{eff}^i(\bar{s}_k) , \quad 2.24$$

with $H_{eff}^i(\bar{s}_k)$ being the effective field acting on spin i on behalf of equa-

tions 1.12 and α being the damping parameter of the LLG equation 2.10. The corrector step replaces the effect $a_i(\vec{s}_k)$ of the LLG equation with the average between the previous \vec{s}_k and the predicted \tilde{s}_k :

$$\vec{s}_{k+1}^i = \vec{s}_k^i + \tau \frac{\vec{s}_k^i + \vec{s}_{k+1}^i}{2} \times a_i \left(\frac{\vec{s}_k + \tilde{s}_k}{2} \right). \quad 2.25$$

As the predictor, this corrector step implicitly conserves spin length and requires the solution of only 3 coupled equations for the three spatial directions. Solvable with a 3×3 matrix, this semi implicit method gains a significant stability advantage while still nearly retaining the speed of a completely explicit iterator.

2.4.1 SIB with perpendicular Spin Transfer Torque

To incorporate the perpendicular thin layer approximation of spin transfer torque (STT) as discussed above in section 2.3.3.3 to the SIB optimiser, one starts with the $a_i(\vec{s}_k)$ utilised in equations 2.23 and 2.25. Instead of employing equation 2.24, as described above, one now proceeds analogous to equation 2.20, yielding

$$\begin{aligned} a_i(\vec{s}_k^i) = & -\mathcal{H}_{\text{eff}}^i(\vec{s}_k) - \alpha \vec{s}_k^i \times \mathcal{H}_{\text{eff}}^i(\vec{s}_k) \\ & - a_j \alpha \hat{S}_p + a_j \vec{s}_k^i \times \hat{S}_p. \end{aligned} \quad 2.26$$

This extends the effect a_i of the LLG equation onto spin i by the two bottom terms, describing the effect of the chosen approximation for spin transfer torque. With the approximation of a constant polarisation direction of the spin polarised current \hat{S}_p constant, the computational cost of these additions is linear with the number of spins and therefore negligible.

In this chapter framework for atomistic spin dynamics, [Spirit](#) is introduced. Starting by outlining the design principles and general structure, the chapter continues to explain the provided visualisation options and the [GUI](#). The live options to control simulation parameters on the fly are described and the python interface to script a simulation run is covered. The section finishes with an summary of features which are under construction or planned in the future.

The starting point for [Spirit](#) was juSpinX [40], a Fortran implementation of atomistic spin dynamics, written by David Bauer, Nikolai Kiselev and Gideon Müller at the Quantum Theory of materials institute of the FZ Jülich. In the scope of this work that code was translated and adapted to C++, forming the basis for the core library and extended with the above described spin transfer torque into the [SIB](#) iteration method by the author. The code was then completely restructured, modularised and extended in collaboration with Gideon Müller. Subsequently the QT [GUI](#) was drafted by Gideon Müller and implemented in a joint effort with the author. The initial visualisation was written by Gideon Müller and Ingo Heimbach (PGI/JCNS at FZ-Jülich) and then replaced by the gl library implemented by Florian Rhiem (PGI/JCNS at FZ-Jülich) who also realised the Web UI and the WebGL code. The Python UI was implemented by Gideon Müller and extended by the author.

Currently the code is Open Source under the MIT License and can be found on [github](#) [39].

The main goal of [Spirit](#) is offering unrivalled modularity, flexibility and operating system independence while still retaining very good performance for atomistic spin dynamics. This is achieved by separating the simulation problem into different and nearly independent parts as in this document, providing clean interfaces for their interaction.

The main part of the code is the core library which can be directly controlled with a [GUI](#) based on the QT application framework [64] or by a classical main file at build time, or alternatively can be compiled into a

library, which in turn can be accessed by e.g. Python.

The first step, in the [GUI](#) done implicitly, is initialising the first spin system. This spin system class is a container holding all the information about the physical system. It contains the following notable members:

- Geometry class, describing the positions of the different spins in relation to each other.
- Spins vector, holding the current orientations of the magnetic moments.
- Hamiltonian class describing the energy and effective field calculation rules as described in [chapter 1](#).
- Present value of Energy and effective field vectors.
- LLG parameter class containing the LLG parameters like damping, temperature or timestep.

After generating this data either from reading a configuration file, user input or by default values, usually an optimiser is initialised. Utilising [SIB](#) for this work, the optimiser knows how to iterate the system e.g. according to the predictor, corrector scheme of the [SIB](#) described in [section 2.4](#).

The optimiser in turn creates a method class, representing the iteration method which is going to be applied. For this work this is generally the atomistic spin dynamic [LLG](#) equation as introduced and motivated in [section 2.2](#), but also geodesic nudged elastic band ([GNEB](#)) and under current development minimum mode following ([MMF](#)) are possible choices. This method can either depend on one system alone like the [SIB](#) or have inter-system forces to simulate transitions between two different states, as is commonly the case with [GNEB](#).

The framework can hold multiple spin systems simultaneously with their respective methods and optimisers. This enables the user to simulate different parameter sets in parallel or calculate energy transitions with the [GNEB](#) method. Copy constructors are provided, which facilitate the simple and fast copying of existing systems as opposed to initialising them anew from file.

Summing up the core library, [figure 3.1](#) is a helpful visualisation, depicting how the frameworks iterates a given system. Assuming initialisation is completed, the control flow (depicted in red) calls the "Iterate" function of the optimiser. The optimiser picks the requested method, which continues by referring to the spin system and its Hamiltonian to get the effective field. Subsequently the method modifies this effective field according to

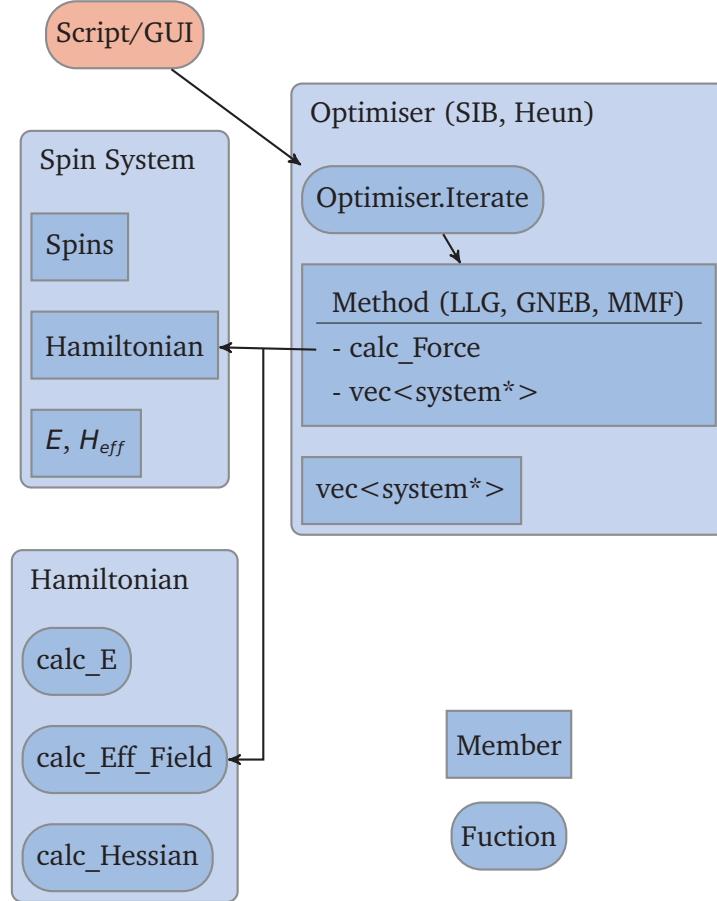


FIGURE 3.1 – Call structure of iterations in the core library

After initialisation and setup, the script or GUI calls the iterate function of the optimiser, which then uses the respective method to get an effective field or energy and then iterate the system according to e.g. the **SIB** method described in [section 2.4](#).

the method specification and returns it to the optimiser. The optimiser in turn performs the iteration and subsequently updates the spins array of the spin system.

While this does sound convoluted and slow, a very fast implementation can be achieved. Avoiding copying of data with references and pointers and employing techniques like C++11's perfect forwarding, a very high efficiency is achieved with only a minimal amount of computational overhead. The huge advantage of this structure, however is exactly the modularity. If a different Method is required to simulate certain problems or investigate a specific system, the code can be extended simply by adding another method class. Originally only containing **LLG** and **GNEB** this process is momentarily happening with the **MMF** method, which is currently being developed and implemented.

Analogous is the addition of different optimisers and Hamiltonians. In this work the "isotropic Hamiltonian" was employed, exploiting the symmetry of the simulated lattice and therefore utilising equations 1.1, 1.2, 1.3, 1.4, 1.5, 1.8, 1.9 and 1.10, quantifying the interactions. However utilising the "anisotropic Hamiltonian", one can also abolish the notion of a symmetric, regular lattice and define interactions between particles and with external fields completely arbitrarily. Giving flexibility in this regard, [Spirit](#) is easily expandable by further Hamiltonians.

In a nutshell the library is completely modularised and extendable in a very simple way. Together with an open source code, this gives the opportunity to collaborate with many people with fixed interfaces where everybody can attach onto and extend by writing their own Hamiltonian, methods or optimiser.

After describing the functionality and building blocks of the core library the following segment is concerned with the control flow of the code. This can be done in different ways as described above, the following two sections, however, will focus on the QT [GUI](#) variant where live control of the simulation parameters and the visualisation are described.

3.1 LIVE CONTROL OF SIMULATION PARAMETERS

Aiming for flexible and different control flows, governing the same set of high-performance calculations, the core of [Spirit](#) is in the form of a library. An application programming interface ([API](#)) is implemented and currently polished, which exposes the important public functions in a comprehensible, easy-to-use fashion. This splits the computational effort into control on the one hand which does not have to be high-performing and can also be localised to specific platforms or infrastructure and core on the other hand. The core as described above is a closed set of code, which can be compiled for every major operating system, does not require any runtime overhead and is highly performance optimised. This gives the flexibility of using the same set of functions and performing the exact same calculations, independent of the platform or the Interface.

The [GUI](#) version of the code gives the user the most options to control the simulation in real time. The main window is depicted in [figure 3.2](#), featuring a big visualisation pane showing the current state of the selected spin system. Furthermore a control bar is at the bottom, enabling the start and stop of simulations, control over which method and optimiser are employed, options to save current system energy and reset the camera along the main axes. The right hand side is by default home of the control and

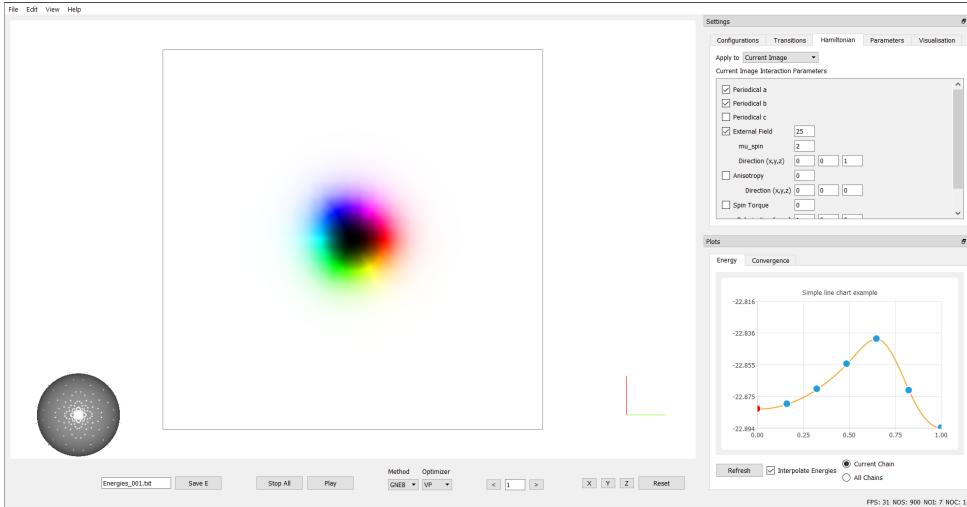


FIGURE 3.2 – Spirit’s GUI developed with QT

The main window of the code shows a visualisation of the current spin system while providing numerous control settings and plotting capabilities on the right hand side. Giving the option to pick the employed method and optimiser, the control bar at the bottom also enables the user to reset camera angles, start and stop the simulation and switch between different spin systems.

plot widgets, which can be resized, moved and undocked in an intuitive fashion.

The shown GUI accesses the API of the simulation core to modify values and properties of the respective classes on the fly. While this is principally also possible mid-iteration, a cleaner behaviour is generally achieved by pausing the iteration first, changing variables and then restarting. Otherwise the first half of the iteration is executed with different parameters than the second one.

The Hamiltonian can be controlled with the respective tab in the settings widget as depicted in figure 3.3 on the left side. All the interaction parameters of the extended Heisenberg model introduced in chapter 1 can be changed live, given that the respective interaction is assumed to be isotropic by the Hamiltonian. This offers the unique opportunity to rapidly experiment with a huge parameter space extremely quickly. Both for research and especially teaching this function is invaluable: Being able to immediately observe the effect of a stronger external field or a different anisotropy, an strong intuition can be developed quickly by observing how the system reacts when different parameters are manipulated.

On the right side of the same figure 3.3 parameters tab is shown, which give control over the parameters of both the **LLG** equation and the **GNEB**

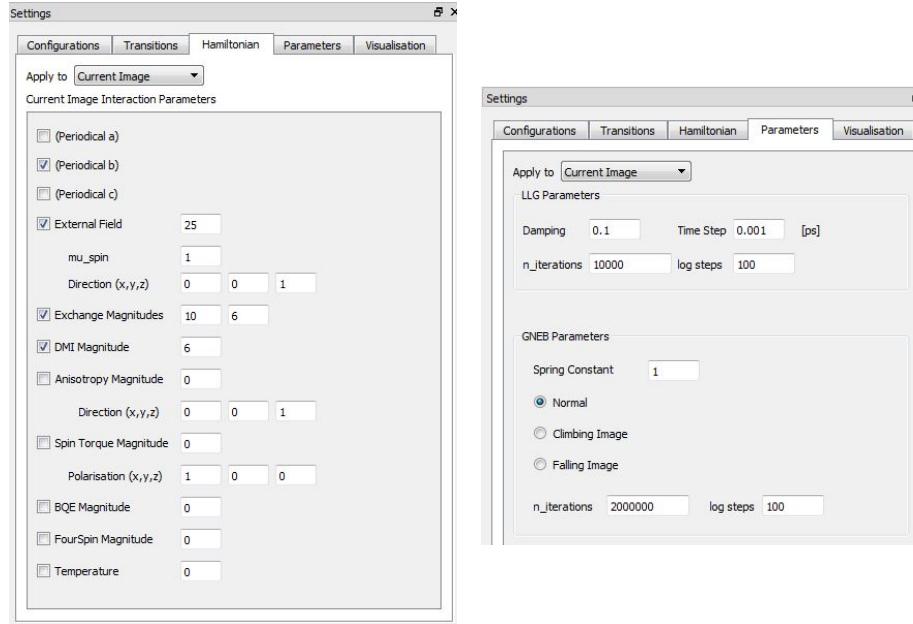


FIGURE 3.3 – Hamiltonian and LLG simulation settings

The left hand image shows the Hamiltonian tab. It enables the user to control every aspect of their, in this case isotropic, hamiltonian by allowing input for the constants of [chapter 1](#). On the right hand side, the simulation parameters of both [GNEB](#) and [LLG](#) are controllable.

calculation method. Here the time step, damping, the number of iterations executed per press of the "play" button and the frequency, how often logs are dumped as files, can be controlled for the [LLG](#) method. For the [GNEB](#) the currently viewed system (called "image" by the code) can be set to behave as either normal, climbing or falling image, the spring constant can be modified and the iteration numbers controlled.

Being able to control all the above mentioned parameters, still the spin system is till now only modified by the iteration algorithms. The configuration tab depicted on the left in [figure 3.4](#) gives the user to initialise different structures in the system. Homogeneous alignment, randomisation, creation of domain walls, spin spirals and skyrmions are facilitated by this pane. With help of parameters the size, position and further characteristics of these objects can be specified and upon execution the spin in the affected region are aligned accordingly. It should be noted that changing configuration of the system mid-iteration is inadvisable. While the code will not throw exceptions or crash, the behaviour is undefined due to the system being set into a different state while an iteration is running in a seperate thread. A blocking call is not implemented at this point, so whatever is written into memory last will stay there, sometimes resulting in setting of only half the intended structure or having no effect at all.

3.2 VISUALISATION

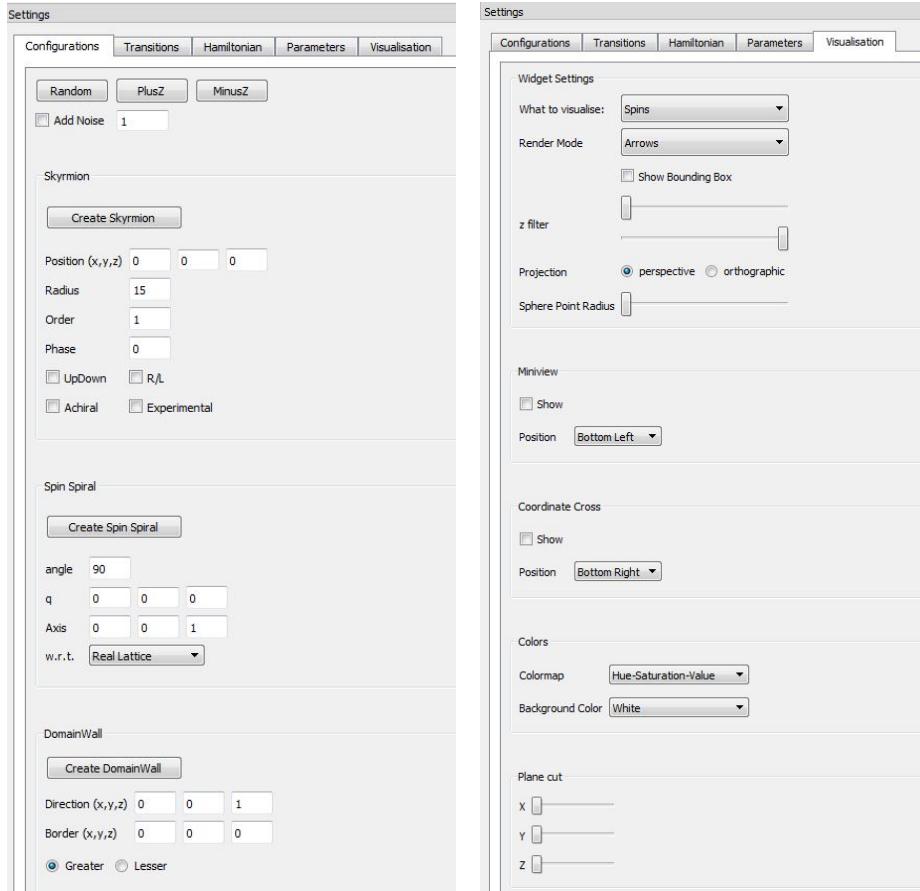


FIGURE 3.4 – Configuration and visualisation setting

The left hand image shows the configurations tab. It provides methods to orient the magnetic moments in a certain way. Homogeneous orientations, domain walls, skyrmions or spin spirals are currently supported and easily configurable. On the right side are the visualisation controls. The spins can be depicted as either arrows or a surface interpolation, a z –filter can be applied, different colour schemes are selectable and projection modes and different inlays are switchable.

3.2 VISUALISATION

This section briefly discusses the employed OpenGL visualisation of [Spirit](#). Having modularity, flexibility and in-situ control as important design principles, missing a live visualisation is not an option. Originally implemented with the visualisation tool kit VTK [65], the visualisation was rewritten completely in OpenGL.

From the computational side, it is independent from the core library, executed on a separate thread and on the graphics card if possible. While

accessing the same data as the iteration in real time can not be without performance impact, this interaction between visualisation and iteration was narrowed down to an irreducible minimum by only reading the same memory block. Therefore the performance impact of the visualisation onto the iteration speed is negligible.

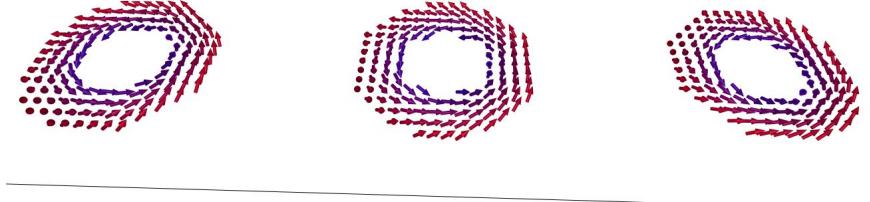


FIGURE 3.5 – Visualisation with Red/Blue colormap and z –filter

The z –filter limits drawing of spins to the ones with a certain range of z –values in this case utilising the red/blue colormap. The bounding box of the system is activated to see the simulated nanostripe, in which three skyrmions are easily visible.

As depicted in [figure 3.4](#) there are various visualisation options. While currently only able to visualise the spins and their respective magnetic moments, effective field and force visualisation are planned. Being able to chose between different backgrounds, projections and colour maps, one can tune the contrast of the picture nicely to visualise structures in a fitting way. Furthermore a z –filter is available as is shown in [figures 3.5](#) and [3.6](#), where only the spins with a certain z –value are drawn in the visualisation.

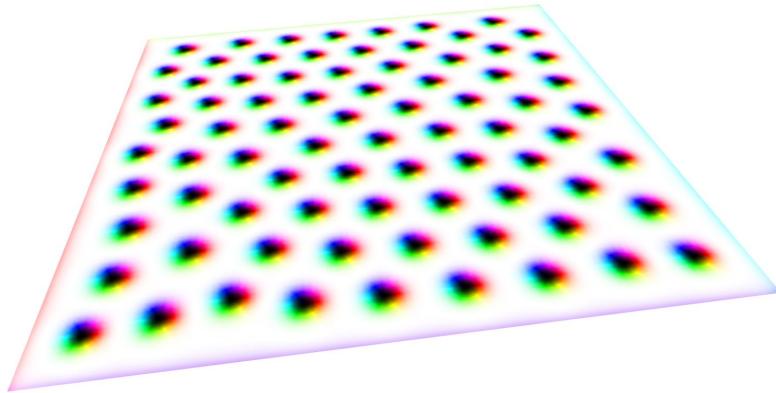


FIGURE 3.6 – Visualisation with surface interpolation

The surface interpolation does not visualise single spins as arrows, but draws the complete plane, interpolating the value at every point from the neighbouring spins and then applying a colour map. This picture is taken from the web version of [Spirit](#), depicting a 100×100 grid with a skyrmion lattice with open boundary conditions.

Another feature is the surface interpolation. Especially useful for larger two dimensional systems, this visualisation does not draw single spins, but a continuous plane. This plane can be filtered by the above described z -filter and is coloured by the chosen colour map. Surface points between the spins are interpolated linearly between the neighbouring spins. This results in a smooth easy-to-read visualisation of a planar system, where structures like skyrmions or domain walls are immediately apparent.

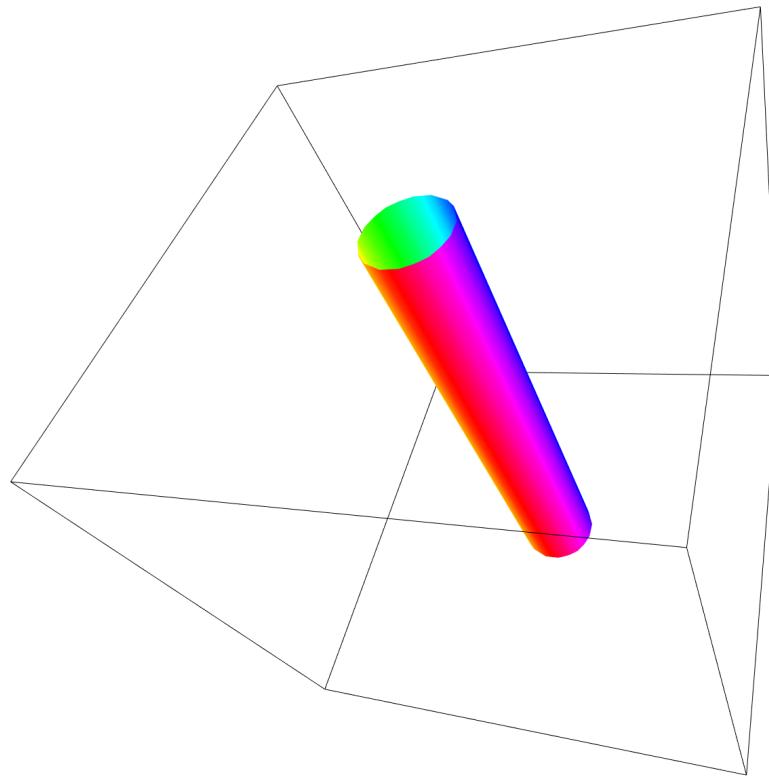


FIGURE 3.7 – 3D skyrmion with isosurface visualisation

A $30 \times 30 \times 30$ spin system with a skyrmion is used to demonstrate the isosurface visualisation. The drawn cylinder is the $z = 0$ plane of the skyrmion.

The latest addition to the different visualisation is the three dimensional iso-surface. Depicted in [figure 3.7](#) this visualisation option facilitates the powerful visualisation of three dimensional objects. The inner working is a mix between surface interpolation and the z -filter. It looks for a specific z -value of spins interpolates between nearest neighbours to find the exact point where the z -value would be e.g. 0. This is repeated for all spins and the interpolated points with a certain z -value are subsequently used to triangulate a surface with a given colour map, depending on the x - and y -value.

As exemplified in [figure 3.8](#) this is an invaluable asset to visualise, charac-

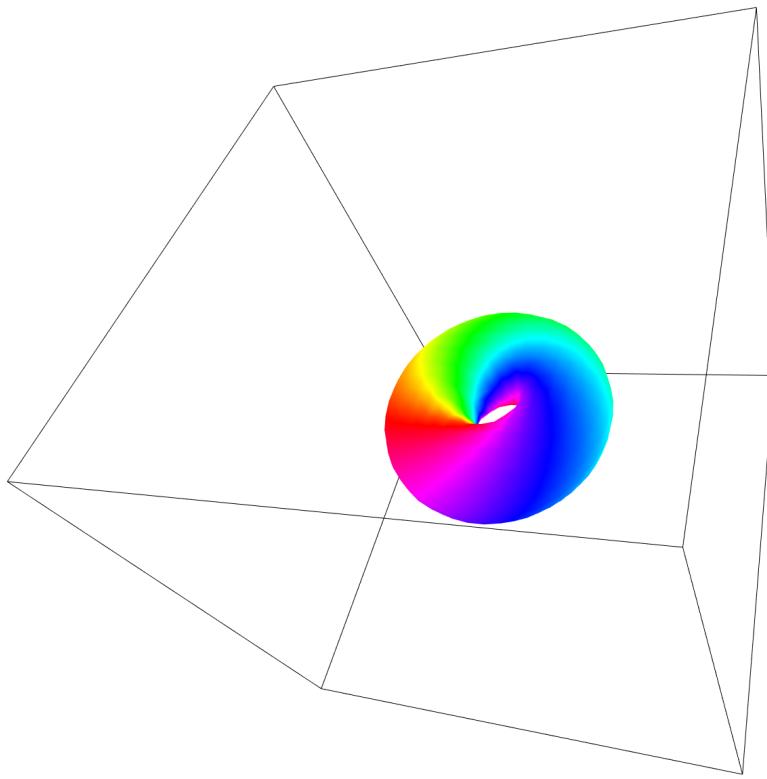


FIGURE 3.8 – Hopfion in isosurface visualisation

A Hopfion with one rotation is visualised in a $30 \times 30 \times 30$ spin system. Depicting the $z = 0$ plane, the isosurface visualisation especially shines in more complex situations like these, where it can provide an intuitive understanding of the non-trivial magnetic structure.

terise and investigate the three dimensional structure of non-trivial magnetic states. Especially for topologically complex objects this is essential, since their structure might not easily be visible in a bulk arrow or surface cut visualisation.

3.3 PYTHON SCRIPTING OF SIMULATIONS

While the [GUI](#) with nice visualisations is a powerful tool and very well suited for exploratory or qualitative investigations, it is limited in terms of repetitive quantitative measurements. This issue can be resolved by using the Python scripting interface. The [API](#) of the core library is completely usable in a python script, without having to recompile anything. The library is compiled once and then subsequently only this binary is accessed by the python scripts, significantly reducing unnecessary compile time and boosting productivity.

Furthermore, utilising the exact same core as above, the user can first explore the system with a [GUI](#) and do some qualitative work, then the script his quantitative experiment in python and extract hard data from there. Adjusting the logging parameters of the core correctly, the initial state, the final result and everything in between can be automatically written into a configuration file, which is easily accessible with the provided [GUI](#). By simply importing the state of the system with the "File" menu, the user can track progress of the system during the simulation, produce figures in a very straightforward way and easily restart or change any point of the simulation, where e.g. unexpected behaviour occurred.

3.4 CURRENT DEVELOPMENT AND VISION

While already being a valuable tool in it's current state, [Spirit](#) is constantly extended and improved. Some features in current development are the addition of the [MMF](#) method in addition to the existing [GNEB](#) in order to extend the capabilities of exploring energy landscapes not only in respect to a predefined and sometimes very limiting guess. Furthermore the [API](#) is currently finished to finalise the complete separation and independence of the core simulation modules and the control flow of the [GUI](#) or scripting. Another very interesting project is the web variant of [Spirit](#): In order to achieve this, the C only wrapped version of the core library was compiled to LLVM bitcode and from there compiled into javascript with emscripten [66]. While currently not having a complete integration for touch features, the potential of running a simple atomistic spin dynamics simulation within a browser and even on a mobile phone is enormous. While maybe not suited for quantitative application, the potential to demonstrate results, code capabilities and features can not be dismissed. Not limited to this, features like such are also essential in scientific outreach and make the code usable for teaching and learning. Being able to simply run it on the fly without the need of installation or compilation, facilitates the use as a teaching tool, enabling students to play with system parameters and gain an intuitive understanding of the different interactions, their interplay and the underlying physics.

Summarising, the atomistic spin dynamics simulation framework [Spirit](#) is a comprehensive and modular code, which still retains a high performance by separating control and simulation program flow. Changing the project's nature to open source, the hope for the future is, to increase the base of active contributors and starting collaborations with people working on similar problems in the same or neighbouring fields of physics. Being completely modular and easily extendable, the current structure is specific-

cally designed to enable this process, encourage people to implement their own ideas into the framework and therefore extend the capabilities and usability of [Spirit](#).

After describing the features and functionality of the simulation framework [Spirit](#), in this section the code's SIB method is first checked in respect to the time axis and then subsequently used to move a domain wall and Skyrmion with help of an external magnetic field and spin polarised current.

4.1 TIME TEST

The first step of using the code, is to prepare the code for quantitative analysis of dynamics simulations. Until now it was only used for static calculations like stability and energy calculations. Therefore it can be assumed to be accurately calibrated for the different terms of the [LLG](#) equation in effective field and energy calculations, both in absolute values as well as in relation to each other. However, for investigating dynamic scenarios, the time step τ of the SIB method which can be input in picoseconds (see [figure 3.3](#) on the right side) has to be tested to ensure the code is working quantitatively correct in that regard.

In order to do this, a singular spin pointing along the x -direction is considered to be exposed to an orthogonal magnetic field along the z -axis without damping. Simulating the system with the SIB method as described in [section 2.4](#), the electron spin precesses around the magnetic field as expected according to classical electrodynamics and Lamor precession. The frequency of this precession is called the Lamor frequency and is given by

$$\omega = -\gamma B , \quad 4.1$$

where ω is the angular frequency, B the magnitude of the magnetic field and γ the gyromagnetic ratio. Substituting in the free electron gyromagnetic ratio γ_e and a magnetic field of $B = 1\text{T}$ this yields

$$\omega \approx 1.7609 \cdot 10^{11} \frac{\text{rad}}{\text{s}} . \quad 4.2$$

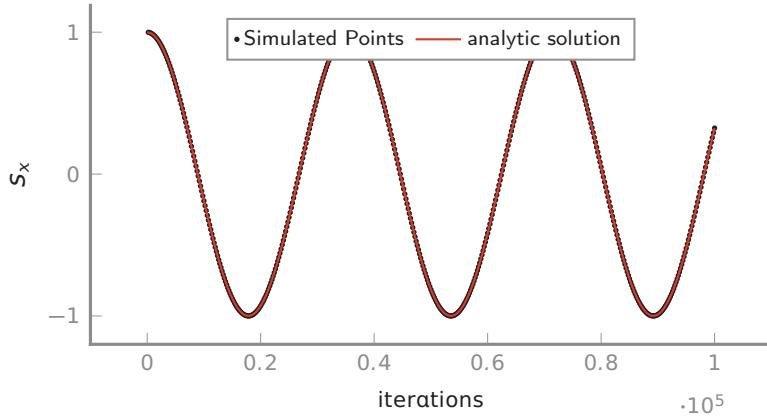


FIGURE 4.1 – Simulated Lamor precession of a single spin

A single spin is precessing in a perpendicular magnetic field of $B = 1\text{ T}$ fitted to a $s_x(t) = \cos(\omega \cdot t)$ with $\omega = 1.7609 \cdot 10^{-4} \frac{\text{rad}}{\text{iteration}}$

Using a time step of $\tau = 1\text{ fs}$ per iteration, the [SIB](#) yields the expected precession. The x -component of the spin s_x is plotted in [figure 4.1](#) and fitted to $s_x(t) = \cos(\omega \cdot t)$, yielding

$$\omega \approx 1.7609 \cdot 10^{-4} \frac{\text{rad}}{\text{iteration}} = 1.7609 \cdot 10^{11} \frac{\text{rad}}{\text{s}} . \quad 4.3$$

This reproduces the analytical result of the Lamor precession and therefore validates the quantitative accuracy of the time step in the code.

4.2 EXAMPLE OF DOMAIN WALL MOTION

Before moving on to skyrmions, the somewhat simpler example of moving a domain wall is studied. To this effect, a basic cubic lattice of 3 by 200 spins is initialised with open boundaries. Furthermore only a simple approximation is considered in this case, resulting in a total Hamiltonian which consists of equations [1.1](#), [1.2](#) and [1.4](#):

$$\mathcal{H} = -\mu_s H \sum_{i=1}^{N_S} \hat{H} \cdot \hat{n}_i - K \sum_{i=1}^{N_S} (\hat{K} \cdot \hat{n}_i)^2 - \sum_{\langle ij \rangle} J_{ij} \hat{n}_i \cdot \hat{n}_j , \quad 4.4$$

only retaining (in this order) the terms for external magnetic field, uniaxial anisotropy, nearest neighbour and next-nearest neighbour interaction. The chosen parameters are displayed in [table 4.1](#) with $J_{1,ij}$ denoting the nearest neighbours exchange constant and $J_{2,ij}$ the constant for next-nearest neighbours.

In the whole simulation, a time step of $\tau = 2\text{ fs}$ and a damping of $\lambda = 0.1$

Parameter	Value	Units	unit vector	direction
μ_s	1.0	μ_B	\hat{H}	(0, 0, 1)
$J_{1,ij}$	10.0	meV	\hat{K}	(0, 0, 1)
$J_{2,ij}$	6.0	meV		
K	2.0	meV		

TABLE 4.1 LLG Parameters for domain wall simulation

is used. To start off, the spins are initialised the following way

$$\hat{s}(x) = \begin{cases} (0, 0, 1) & \text{for } x < 20 \\ (1, 0, 0) & \text{for } x = 20 \\ (0, 0, -1) & \text{for } x > 20 \end{cases}, \quad 4.5$$

with x being the x -component of the spin position in atomic distances. This yields the most basic variant of a Neel type domain wall, which is subsequently relaxed with 10^4 iterations (corresponding to 20ps with the above mentioned time step) as shown in [figure 4.2](#)

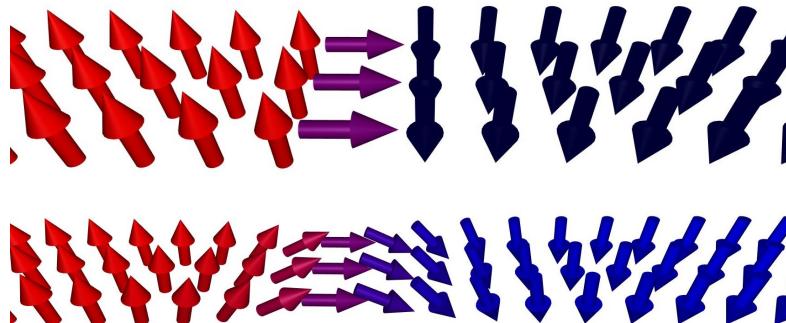


FIGURE 4.2 – Initial state and relaxed Neel type domain wall

The system is initialised according to [equation 4.5](#) (top picture) and then relaxed with SIB method to get the final starting point (bottom picture) for the domain wall movement simulation.

4.2.1 Domain wall motion induced by an applied magnetic field

After the domain wall is relaxed, the system is saved and an external magnetic field employed along the $+z$ direction. This energetically favours the upwards pointing spins, pushing the domain wall to the right hand side ($+x$ direction). While this happens, the spins undergo Lamor precession as

described in the [LLG equation 2.10](#) around the anisotropy and magnetic field axis, yielding a precessing and forward moving domain wall. This is depicted in [figure 4.3](#)

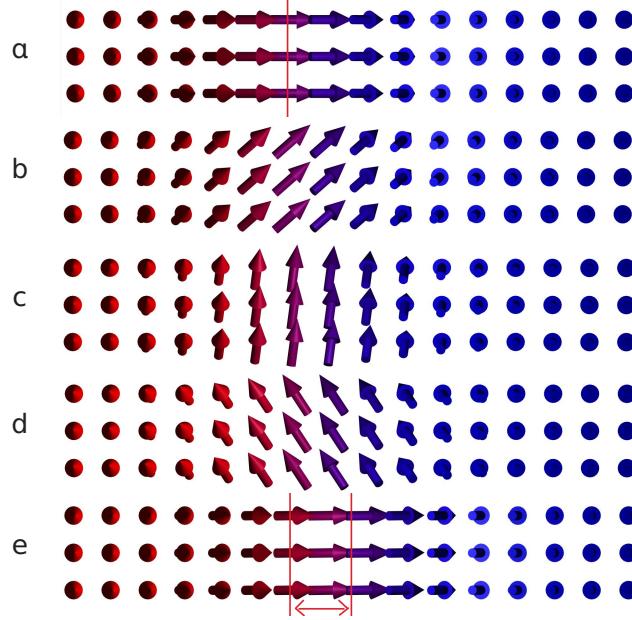


FIGURE 4.3 – Precessing steady motion of the domain wall induced by an external magnetic field

The external magnetic field is pointing along $+z$. The system is initialised at a with the above described Neel domain wall in a field of $B = 2\text{T}$ and between each step iterated 10^3 times with a time step of 2fs from a till d. Between d and e there are a differing 5750 iterations to showcase the forward translation of the domain wall, after one full Lamor precession rotation in comparison to the initial state.

To investigate the forward motion of the domain wall, the same initial domain wall is copied multiple times and simulated with different external magnetic fields. After $5.5 \cdot 10^5$ iterations the domain wall positions were evaluated and plotted against the magnetic field of the corresponding system in [figure 4.4](#)

The resulting plot shows a linear dependence of domain wall speed to external magnetic field. With the total iteration time of 1.1ns , one can transfer the linear slope to a velocity depending on the external magnetic field.

$$v_x(B_{\text{ext},z}) = B_{\text{ext},z}[\text{T}] \cdot \frac{43.345}{1100} \left[\frac{a}{\text{T} \cdot \text{ps}} \right], \quad 4.6$$

with a being the lattice constant of the simulated simple cubic lattice.

Comparing this linear dependence to the results obtained by [Mougin et al. \[10\]](#) or [Glathe et al. \[11\]](#), one might miss the described walker breakdown

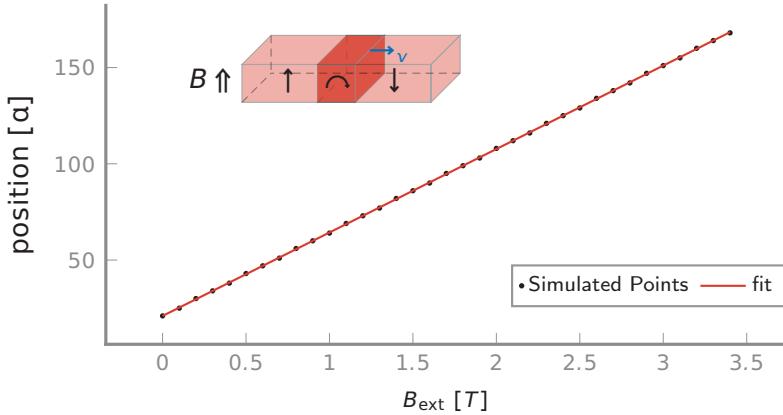


FIGURE 4.4 – Domain wall motion in an external magnetic field

The domain wall is initialised at $x = 21$ and then simulated for different magnetic fields along $+z$. The position in lattice constants after $5.5 \cdot 10^5$ iterations with $\tau = 2\text{fs}$ is plotted against the magnetic field. The datapoints are fitted, yielding a slope of 43.345 ± 0.048 and an intercept of 20.98 ± 0.09

of the domain wall motion. This is due to the fact that they, additionally employ a demagnetisation torque

$$\Gamma_{H_d} = 4\pi M_s^2 \begin{pmatrix} 0 \\ (N_y - N_x) \sin \theta \sin \varphi \cos \varphi \\ \sin \theta \cos \theta [N_z - N_y \sin^2 \varphi - N_x \cos^2 \varphi] \end{pmatrix}, \quad 4.7$$

which models the demagnetisation field \vec{H}_d within the domain wall (using spherical coordinates), where N_i are the demagnetisation factors and M_s is the saturation magnetisation. In case the demagnetisation torque was to be applied, two regimes would emerge:

- In a low field environment, the demagnetisation and damping term dominate the movement, yielding a simple forward moving domain wall that does not precess.
- Increasing the magnetic field to and above the walker field, the Lamor precession term grows until it competes with the damping and demagnetisation. While a forward motion with Lamor precession can be observed in this case, unlike in this work, that motion is characterised by deformations of the domain wall and an oscillatory back-and-forth motion with a net forward effect.

An important notion is, that the measure velocity per Tesla of external field of these two regimes differ significantly. A more in-depth discussion of this phenomenon can be found in [Mougin et al. \[10\]](#) and experimental data e.g. by [Glathe et al. \[11\]](#).

However, since this work is not only concerned with domain walls, a structure specific term like that was not implemented. Omitting this term and the damping it provides, one expects a domain wall to exhibit Lamor precession whilst being moved by a magnetic field with the velocity depending linearly on the applied field, as was observed in the simulations.

4.2.2 Motion induced by a spin polarised current

After the simple case of movement induced by an external magnetic field, in this section a spin polarised current will be employed as the driving force for domain wall motion. As described in section [section 4.2](#), the domain wall was initialised as a Neel type wall and relaxed with the SIB with only anisotropy and next-nearest next neighbour interaction present (compare to [equation 4.4](#) without the external magnetic term ($H = 0$)). Time step, as well as number of iterations and all other parameters, except the external magnetic field, are equal to the ones used in [section 4.2.1](#) and found in [table 4.1](#).

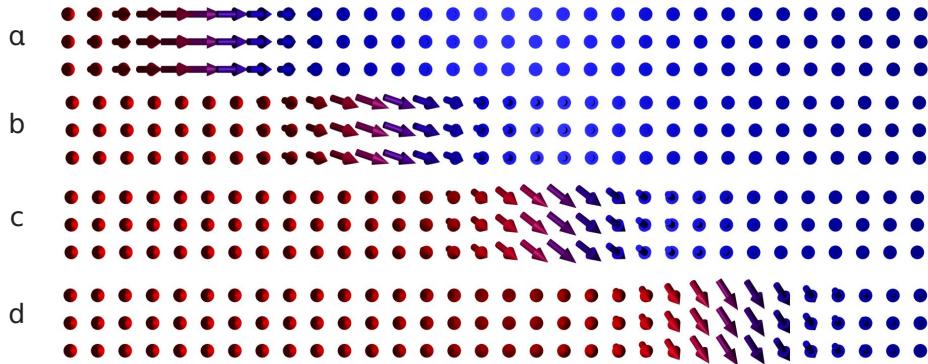


FIGURE 4.5 – Precessing movement of the domain wall pushed by a spin polarised current

The system is started at a with the above described Neel domain wall and then exposed to a current with $a_j = 0.1$ which is spin polarised in the $-z$ direction, showing propagation of the domain wall in x direction with simultaneous Lamor precession around the uniaxial anisotropy. The snapshots are taken $5 \cdot 10^3$ iterations apart with a time step of $\tau = 2\text{fs}$, starting at a through to d.

After relaxation the spin polarised current is applied, as described in [section 2.4.1](#), perpendicular to the thin stripe with spin polarisation $(0, 0, -1)$. The resulting movement of the domain wall can be seen in [figure 4.5](#) and qualitatively behaves similar to the domain wall pushed by an external magnetic field. While the wall is moving forward, it exhibits Lamor precession around the out of plane anisotropy.

Looking closer at this and comparing this to [section 4.2.1](#) and especially

figure 4.3, one notices two things:

- The Lamor precession is a much weaker effect, compared to the domain wall moved by the external magnetic field.
- The spins precess clockwise here, while above they precess counter-clockwise.

The first point is trivial, since the transition is here dominated by the spin polarised current, which has no Lamor precession associated with it, while the out-of-plane anisotropy only plays a minor role, causing the observed weak precession (compare figure 4.5). However when moving the domain wall with the external magnetic field, the spins start to precess around the much stronger external magnetic field vector, therefore adding a Lamor precession that grows linearly in strength as the field strength increases. This leads to the second point: Since the anisotropy vector in the region ($+x$), where the domain wall is moving towards, is pointing to $-z$, the Lamor precession observed while moving the domain wall is clockwise (shown from above or from $+z$). However, when the precession around the external magnetic field (pointing towards $+z$) dominates the effect as shown in the previous section, then the observed Lamor precession will inadvertently be counter-clockwise as seen in figure 4.3.

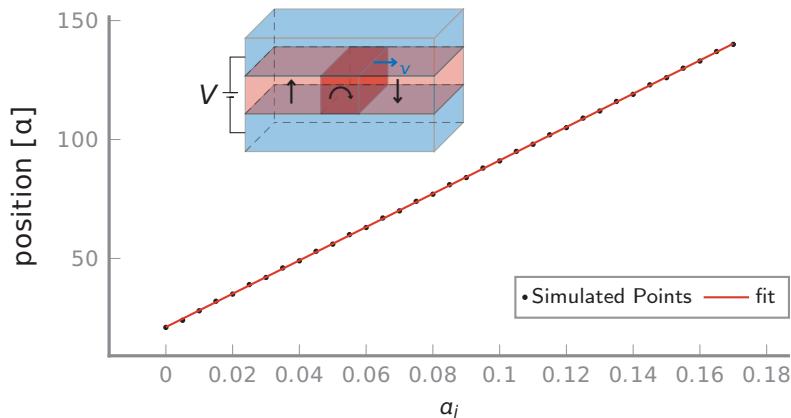


FIGURE 4.6 – Domain wall motion induced by a spin polarised current

The domain wall is initialised at $x = 21$ and then simulated for different spin transfer torque parameters a_j with polarisation pointing towards the $-z$ direction. The position in lattice constants after $5.5 \cdot 10^4$ iterations with $\tau = 2\text{fs}$ is plotted against a_j . The datapoints are fitted, yielding a slope of 700.90 ± 0.93 and an intercept of 21.14 ± 0.09

Summarising the findings so far, Lamor precession around the dominating static field vector (external or isotropic) was observed scaling with the strength of the very same field, as expected. To take a closer look at the

propagation speed of the domain wall, with respect to the current density of the spin polarised current, multiple simulations were run with different spin transfer torque parameters a_j . The result of these simulations are depicted in [figure 4.6](#).

Similarly to the observation for domain wall movement stimulated by external magnetic field, the result here is a linear dependence of the wall's propagation speed in respect to the applied current density. Converting the x position and slope of [figure 4.6](#) into a velocity in terms of lattice constants, this yields:

$$v_x(a_j) = a_j \cdot \frac{700.9}{110} \left[\frac{a}{ps} \right], \quad 4.8$$

with a_j being the spin torque parameter as described in [section 2.4.1](#). As in the external magnetic field induced movement, no Walker breakdown can be observed due to the fact that the structure specific demagnetisation term not being implemented in this structure-independent atomic spin dynamics simulation.

The expected linear behaviour was therefore reproduced in the experiment and proves to be stable up to a spin torque parameter of $a_j \approx 0.465$ for the given system and parameters. Increasing the spin torque parameter further invalidates the question about domain wall movement, since new domains begin to nucleate due to the spin polarised current at that point.

4.3 SKYRMION MOTION INDUCED BY A SPIN POLARISED CURRENT

After the experiments with 180° domain wall motion, now the dynamics of moving a structurally more complex skyrmion via [SPC](#) are explored. The first step towards this goal is choosing a parameter set, where skyrmions can be stabilised easily. The parameters chosen here are based on the work of [Müller](#) [42], adding next-nearest neighbour exchange interaction and disregarding uniaxial anisotropy. The uniaxial anisotropy is disregarded since a preferred axis is already given by the magnetic field and the next-nearest neighbours are included to get a closer approximation with the employed model. This results in a total Hamiltonian consisting of equations [1.1](#), [1.4](#) and [1.5](#)

$$\mathcal{H} = -\mu_s H \sum_{i=1}^{N_S} \hat{H} \cdot \hat{n}_i - \sum_{\langle ij \rangle} J_{ij} \hat{n}_i \cdot \hat{n}_j - \sum_{\langle ij \rangle} D_{ij} \cdot \vec{r}_{ij} \cdot (\hat{n}_i \times \hat{n}_j), \quad 4.9$$

with the corresponding parameters summarised in [table 4.2](#).

Parameter	Value	Units	unit vector	direction
μ_s	1.0	μ_B	\hat{H}	(0, 0, 1)
H	25	T		
$J_{1,ij}$	10.0	meV		
$J_{2,ij}$	6.0	meV		
D_{ij}	6.0	meV		

TABLE 4.2 LLG Parameters for skyrmion simulation

With these settings it should now be possible to stabilise a skyrmion as a metastable state in a local energy minimum. To this end, the "create skyrmion function" of the code is employed, which starts by calculating the distance between the position \vec{r}_i of spin i and the center of the skyrmion \vec{r}_0 in x - and y -direction:

$$r_i = |\vec{r}_i - \vec{r}_0|^{x,y} . \quad 4.10$$

Comparing this with the radius R of the skyrmion

$$\theta_i = \pi \left(1 - \frac{r_i}{R} \right) , \quad 4.11$$

yields the angle θ . The other angle needed is

$$\phi_i = \arccos \left(\frac{\vec{r}_i^x - \vec{r}_0^x}{R} \right) + \varphi \quad 4.12$$

with the additional phase φ and the argument of the \arccos limited to $[-1, 1]$. One can then set the spin i to

$$\vec{n}_i = \begin{pmatrix} \sin(\theta_i) \cdot \cos(\xi\phi_i) \\ \sin(\theta_i) \cdot \sin(\xi\phi_i) \\ \cos(\theta_i) \cdot d \end{pmatrix} , \quad 4.13$$

where $d \in \{-1, 1\}$ is the direction up (down) of the skyrmion and ξ the order parameter. With a radius of $R = 10$, order $\xi = 1$ and a phase of $\varphi = -90^\circ$ this yields a skyrmion as depicted in [figure 4.7](#) on the left side.

This initialisation is close to the local energy minimum, since the skyrmion after SIB relaxation is only marginally different in profile, exhibiting the same size and shape. The radius is approximately 10 atomic distances and therefore for further investigations a strip of $200 \cdot 25$ atoms is simulated with periodic boundary conditions on the y axis. If the skyrmion moves

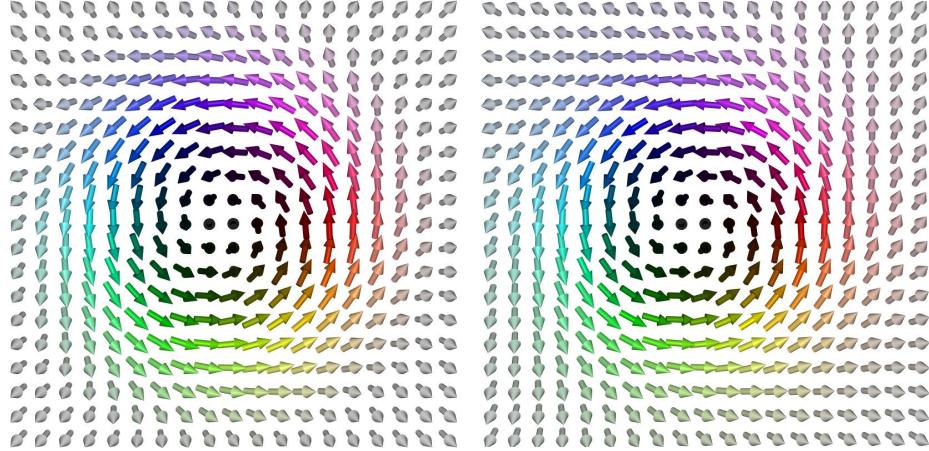


FIGURE 4.7 – Skyrmion initialisation

Skyrmion initialised by the code (left) and after SIB relaxation (right). As is apparent from the near identical images, the skyrmion initialisation is close to the local minimum.

out of the strip (e.g. on the top) it reenters the system on the bottom. Aligning the skyrmion motion along the x axis then allows for easy visualisation of the traversed distance.

Exposing this skyrmion to a spin polarised current, induces a spin torque onto the skyrmion resulting in a linear motion. Choosing a current, polarised along $(-1, 1, 0)$ yields a skyrmion motion nearly parallel to the x -axis. The simulation result of this motion for a spin torque parameter of $\alpha_j = 0.3$ is shown in [figure 4.8](#)

After seeing the skyrmion motion in a qualitative way, now the influence of the spin torque parameter α_j is investigated quantitatively. In order to do this, the skyrmion is initialised and relaxed without any current. Then the system is copied, analogous to the domain wall experiments, and exposed to spin polarised currents with different spin transfer torque parameters. For this quantitative calculation a smaller timestep $\tau = 1\text{fs}$ is chosen for better accuracy. The simulated time frame with $n = 5.5 \cdot 10^4$ iterations is therefore $t = 55\text{ps}$. The result of these simulations are depicted in [figure 4.9](#) with traversed distance plotted over utilised spin transfer torque parameter α_j .

The position of the skyrmion was defined as the spin with the smallest z component and the travelled distance in x and y was projected onto a single axis and subsequently plotted. As in the domain wall experiments, here again a linear dependence of the skyrmion's propagation speed in respect to the spin torque parameter can be found. Converting the fitted

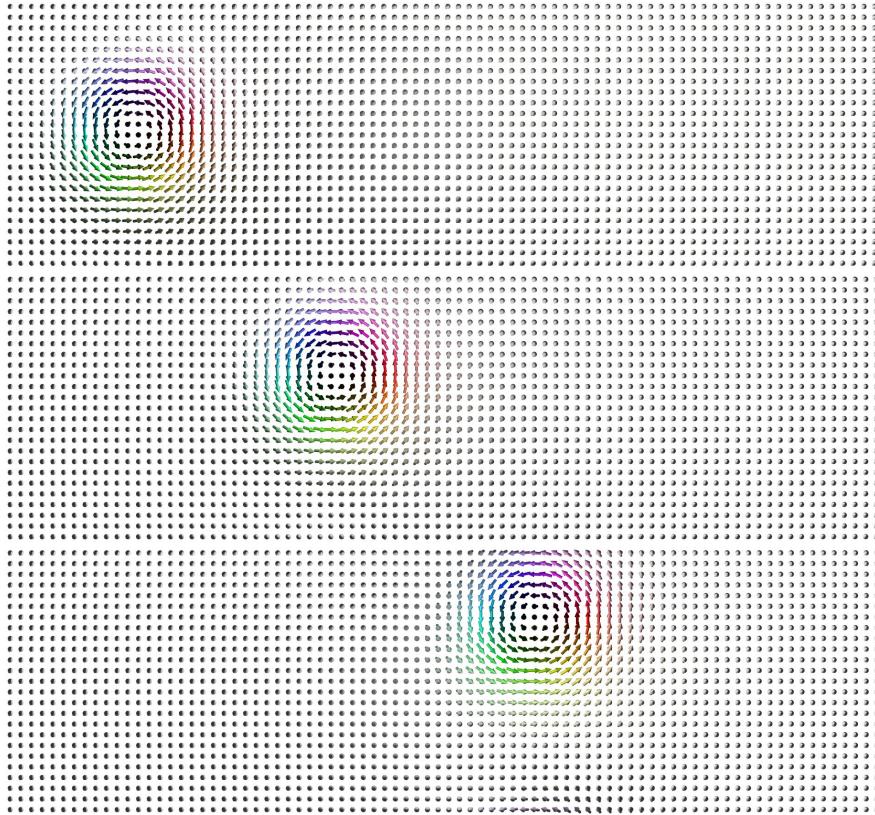


FIGURE 4.8 – Skyrmion motion induced by spin polarised current

The skyrmion is initialised as described by the skyrmion creation function, relaxed by SIB and then exposed to a current with spin polarisation direction $(-1, 1, 0)$ and spin torque parameter $\alpha_j = 0.3$. The images are taken (from top to bottom) with a timestep of $\tau = 10\text{fs}$ and with an equal $n = 500$ iterations between them.

values into a velocity in terms of lattice constants, one ends up with:

$$v(a_j) = a_j \cdot \frac{507.9}{55} \left[\frac{a}{ps} \right] \quad 4.14$$

for the skyrmion motion.

In the simulated system, the skyrmion is definitely stable up to a spin torque parameters of $\alpha_j = 1$ with some static domain-wall-like artifacts appearing on the $-x$ side of the sample. Above that skyrmion destabilises, grows in size and the system overall does not stay in the ferromagnetic state, but starts exhibiting fluctuations and nucleating structures like domain walls.

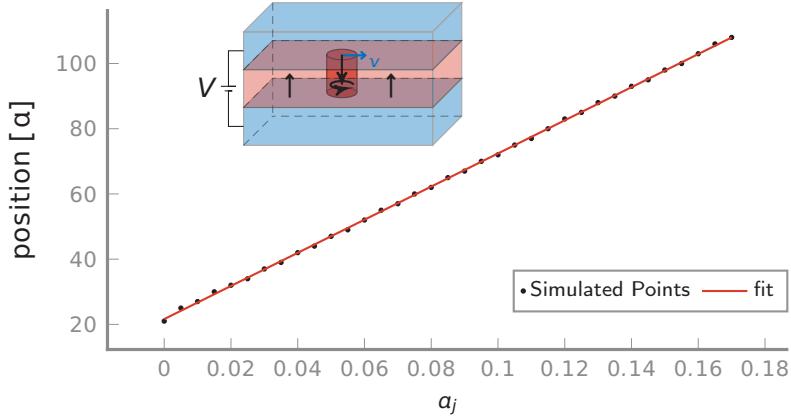


FIGURE 4.9 – Skyrmion motion in respect to spin transfer torque α_j

The skyrmion is initialised at $x = 21$ and then simulated for different spin transfer torque parameters with polarisation pointing towards along $(-1, 1, 0)$ direction. The distance in lattice constants after $5.5 \cdot 10^4$ iterations with $\tau = 1\text{fs}$ is plotted against the current density. The datapoints are fitted, yielding a slope of 507.90 ± 1.24 and an intercept of 21.63 ± 0.12

4.4 SKYRMIONS ON A TRACK-LIKE SAMPLE

After utilising spin polarised current to move a skyrmion in a periodic and therefore practically infinite 2D system, now a true nano-stripe is employed. By picking open boundary conditions on every edge, the sample is limited in size and the skyrmion can be pushed along this "track" as depicted in [figure 4.10](#). The employed parameters are otherwise equal to those of [section 4.3](#). Initialising a skyrmion in the beginning of the sample as described above, the same simulation is executed. An interesting effect is the interaction of the skyrmion with the open boundary on the far right ($+x$) side of the system. With the initial spin transfer torque parameter of $a_j = 0.3$ the skyrmion is repelled, but still manages to be annihilated by the boundary. With a smaller $a_j = 0.2$ this repulsion can not be overcome and the skyrmion stays in the system. Excepting the interaction with the right boundary, the translation of the skyrmion is linear as in [section 4.3](#) and as expected.

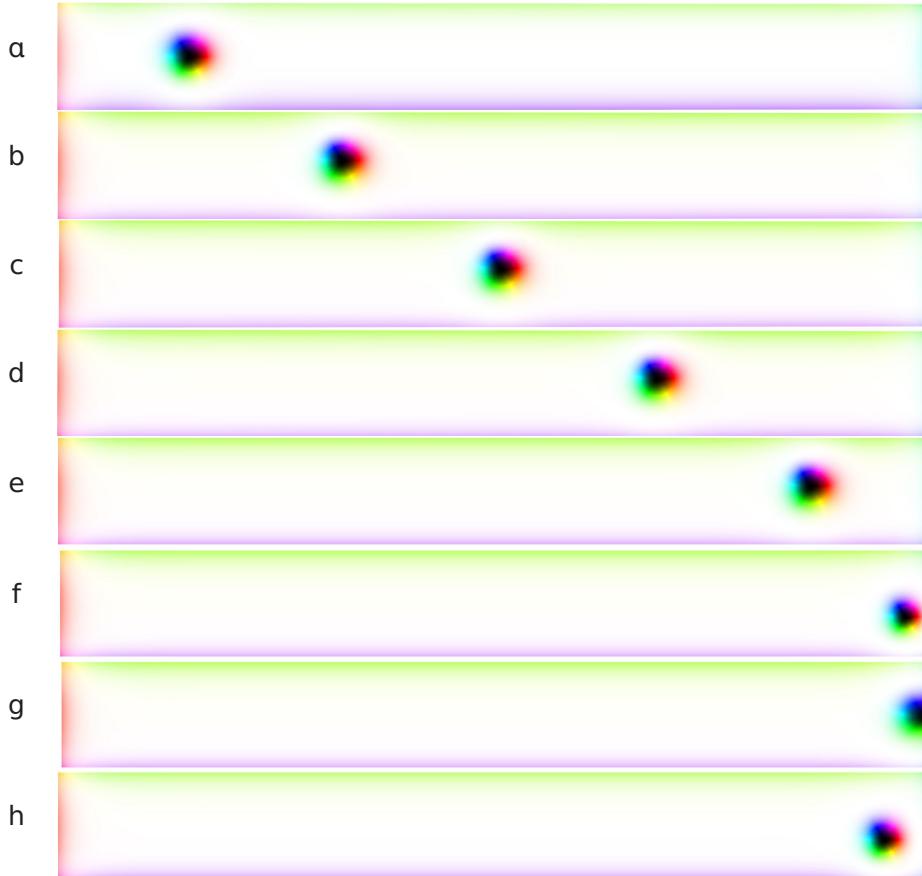


FIGURE 4.10 – Skyrmion on a track-like sample with open boundaries

The skyrmion is initialised and stabilised in **a** with open boundary conditions. Subsequently a spin transfer torque with $a_j = 0.3$ and polarisation direction $(-1, 1, 0)$ is employed like above. Images **b** through to **f** are taken in equal intervals of 10^4 iterations with a time step of $\tau = 10\text{fs}$. As expected the skyrmion moves linearly through the strip. In picture **g** one can see the skyrmion being destroyed at the open boundary. In case of reducing the transfer torque parameter to $a_j = 0.2$ in picture **f**, one ends up with image **h** another 10^4 iterations later. One can see the repulsion of the open boundary upon the skyrmion, which can with the lower spin transfer torque not overcome.

Starting with an [introduction](#) and a placement of the work into the current technological challenges, the employed extended Heisenberg model with respective effective field is described in [chapter 1](#). In [chapter 2](#) the atomistic spin dynamic approach is introduced in contrast to the commonly employed micro magnetic model. For the simulation of skyrmions or other magnetic states localised in a small volume, the continuous field approximation of the micro magnetic model is questionable and therefore the more precise atomistic approach, facilitated by modern day computing power, is chosen. Introducing spin polarised current and the respective effect of spin transfer torque, the chapter closes on the description of the implementation of the method for numerical solution of the [LLG](#) equation for magnetic spin motion.

The following [chapter 3](#) introduces the simulation framework [Spirit](#). The development of the [Spirit](#) framework was one of the main objectives of this work and has been carried out in collaboration with other developers (G. Müller and F. Rhiem) responsible for different aspects of the code. The main contribution of this work was translating the core library from Fortran to C++, implementing the spin transfer torque term into the integration scheme for the stochastic [LLG](#) equation and improving and extending the core library from its initial state onwards. Moreover, significant contributions to debugging and testing of the code were made, some of which are presented in [chapter 4](#) with domain wall and skyrmion motion.

The main goal of [Spirit](#) is providing a comprehensive simulation software that works well across different platforms and provides easy and fast ways of getting numerical insights into a variety of solid state problems via atomistic spin dynamics. [Spirit](#) provides an advanced [GUI](#) with an intuitive in-situ control of simulation parameters and live visualisation. The framework is written in a modular way, enabling the rapid addition of specific features or extensions of the software to accommodate very specific physical problems. For quantitative and repetitive measurements, a

Python scripting interface is provided. Good accessibility and portability is offered by the web version, only requiring a browser to be installed. This combined functional range is unparalleled in atomistic spin dynamics and enables [Spirit](#) to also be used as an educational tool, e.g. enabling students to gain an intuitive understanding of the extended Heisenberg model, the different energy contributions and their respective effect onto a given physical system.

In [chapter 4](#) the accuracy of the time axis of the [SIB](#) is compared to and confirmed with an analytically solvable one-spin system. Domain wall propagation induced by external magnetic field and spin transfer torque is demonstrated with a Neel type domain wall. The contrast to the not-observed walker breakdown is discussed as the effect of the demagnetisation term, which is domain wall specific and therefore not part of this simulation.

Choosing a different parameter set, a magnetic skyrmion of Bloch type is stabilised and subsequently exposed to a spin polarised current in a 2D system. While retaining its structure and stability, it is translated linearly with respect to the spin current density. This is done for periodic boundary conditions and open boundaries. Furthermore the repulsion between the edge of the sample and a skyrmion is shown.

With the functionality and capabilities of [Spirit](#) described in [chapter 3](#) and showcased in a few examples in [chapter 4](#), these are some of the next questions, which can be investigated and answered with the current version of [Spirit](#):

- Skyrmion nucleation and annihilation at boundaries induced by [STT](#), investigation of the corresponding processes and comparison with the results of calculations of the energy barriers with the geodesic nudged elastic band ([GNEB](#)) method.
- Thermodynamic stability of skyrmions at finite temperatures, lifetime and attempt frequency of above mentioned processes.
- Confining a skyrmion in a racetrack-like strip, moving it back and forth in a discretised fashion.
- Influences of system geometry and external factors onto Skyrmion-Skyrmion interaction.

Extending upon the currently given simulation capabilities, the following topics, amongst others, warrant further attention:

- Implementation of gradient [STT](#) method and investigation of Skyrmions in thicker films.

- Investigation of three dimensional magnetic structures like chiral bobbers [67] or hopfions (see [figure 3.8](#)) with respect to spin polarised currents.
- Implementation and investigation of the effects of AC fields and currents onto skyrmion and bobbers.
- Implementation and exploration of thermal gradient effects onto chiral magnets.

Among the above mentioned research topics, the parallelisation of [Spirit](#) is a topic of high importance. While it is currently platform independent and features parallel calculation of multiple systems, this can be further improved upon. A rewrite concerning utilisation of GPU features by the core library is one important step into this direction. Furthermore large systems could be split up into smaller, independently iterateable, sub-systems which are regularly stitched back together, increasing the number of independent threads dynamically with respect to the available CPU and architecture. Coupling this with more extensive profiling and general code optimisations, it should be possible to increase iteration speed significantly.

Increasing the efficiency, the influence of dipole-dipole interaction onto the above described systems could be investigated and quantitatively characterised with reasonable simulation times. Implementation of the [DDI](#) is crucial in judging the accuracy of the model and employed methods and verification, of when it is acceptable to neglect it and when it has to be included.

BIBLIOGRAPHY

- [1] G. Binasch, P. Grünberg, F. Saurenbach, and W. Zinn. Enhanced magnetoresistance in layered magnetic structures with antiferromagnetic interlayer exchange. *Phys. Rev. B*, 39:4828–4830, Mar 1989. doi: 10.1103/PhysRevB.39.4828. URL <http://link.aps.org/doi/10.1103/PhysRevB.39.4828>.
- [2] M. N. Baibich, J. M. Broto, A. Fert, F. Nguyen Van Dau, F. Petroff, P. Etienne, G. Creuzet, A. Friederich, and J. Chazelas. Giant magnetoresistance of (001)fe/(001)cr magnetic superlattices. *Phys. Rev. Lett.*, 61:2472–2475, Nov 1988. doi: 10.1103/PhysRevLett.61.2472. URL <http://link.aps.org/doi/10.1103/PhysRevLett.61.2472>.
- [3] Eric Gaba. Inner view of a 1998 seagate hdd that used parallel ata interface. Online, Wikimedia Commons, Aug 2010. URL https://en.wikipedia.org/wiki/Hard_disk_drive#/media/File:Seagate_ST33232A_hard_disk_inner_view.jpg. CC BY-SA 3.0.
- [4] Andrew Henry Bobeck and Edward Della Torre. *Magnetic bubbles*. Amsterdam ; Oxford : North-Holland Publishing Co. [etc.], 1975. ISBN 0720414636. Bubble domains (BNB/PRECIS).
- [5] S.S.P. Parkin. Shiftable magnetic shift register and method of using the same, December 21 2004. URL <https://www.google.com/patents/US6834005>. US Patent 6,834,005.
- [6] Stuart S. P. Parkin, Masamitsu Hayashi, and Luc Thomas. Magnetic domain-wall racetrack memory. *Science*, 320(5873):190–194, 2008. ISSN 0036-8075. doi: 10.1126/science.1145799. URL <http://science.sciencemag.org/content/320/5873/190>.
- [7] Stuart Parkin and See-Hun Yang. Memory on the racetrack. *Nat Nano*, 10(3):195–198, Mar 2015. ISSN 1748-3387. URL <http://dx.doi.org/10.1038/nnano.2015.41>. Commentary.
- [8] Masamitsu Hayashi, Luc Thomas, Rai Moriya, Charles Rettner, and Stuart S. P. Parkin. Current-controlled magnetic domain-wall nanowire shift register. *Science*, 320(5873):209–211, 2008. ISSN 0036-8075. doi: 10.1126/science.1154587. URL <http://science.sciencemag.org/content/320/5873/209>.

- [9] X.Z. Yu, N. Kanazawa, W.Z. Zhang, T. Nagai, T. hara, K. Kimoto, Y. Matsui, Y. Onose, and Y. Tokura. Skyrmiion flow near room temperature in an ultralow current density. *Nature Communications*, 3(988), Aug 2012. doi: 10.1038/ncomms1990. URL <http://dx.doi.org/10.1038/ncomms1990>.
- [10] A. Mougin, M. Cormier, J. P. Adam, P. J. Metaxas, and J. Ferré. Domain wall mobility, stability and walker breakdown in magnetic nanowires. *EPL (Europhysics Letters)*, 78(5):57007, 2007. URL <http://stacks.iop.org/0295-5075/78/i=5/a=57007>.
- [11] S. Glathe, R. Mattheis, and D. V. Berkov. Direct observation and control of the walker breakdown process during a field driven domain wall motion. *Applied Physics Letters*, 93(7):072508, 2008. doi: <http://dx.doi.org/10.1063/1.2975181>. URL <http://scitation.aip.org/content/aip/journal/apl/93/7/10.1063/1.2975181>.
- [12] A. N. Bogdanov and D. A. Yablonskii. Thermodynamically stable "vortices" in magnetically ordered crystals. the mixed state of magnets. *Sov. Phys. JETP* 68, 101-103, 1989.
- [13] A. Bogdanov and A. Hubert. The properties of isolated magnetic vortices. *Phys. Stat. Sol. (b)* 186, 527, 1994. URL <http://onlinelibrary.wiley.com/doi/10.1002/pssb.2221860223/pdf>.
- [14] I. Kezsmarki, S. Bordacs, P. Milde, E. Neuber, L. M. Eng, J. S. White, H. M. Ronnow, C. D. Dewhurst, M. Mochizuki, K. Yanai, H. Nakamura, D. Ehlers, V. Tsurkan, and A. Loidl. Neel-type skyrmion lattice with confined orientation in the polar magnetic semiconductor gav4s8. *Nat Mater*, 14(11):1116–1122, Nov 2015. doi: 10.1038/nmat4402. URL <http://dx.doi.org/10.1038/nmat4402>.
- [15] X. Z. Yu, Y. Onose, N. Kanazawa, J. H. Park, J. H. Han, Y. Matsui, N. Nagaosa, and Y. Tokura. Real-space observation of a two-dimensional skyrmion crystal. *Nature* 465, 901-904, 2010. doi: 10.1038/nature09124. URL <http://www.nature.com/nature/journal/v465/n7300/abs/nature09124.html?guid=ON>.
- [16] K. Shibata, X. Z. Yu, T. Hara, D. Morikawa, N. Kanazawa, K. Kimoto, S. Ishiwata, Y. Matsui, and Y. Tokura. Towards control of the size and helicity of skyrmions in helimagnetic alloys by spin-orbit coupling. *Nature Nanotech.* 8, 723, 2013.
- [17] T. Adams, A. Chacon, M. Wagner, A. Bauer, G. Brandl, B. Pedersen, H. Berger, P. Lemmens, and C. Pfleiderer. Long-wavelength helimagnetic order and skyrmion lattice phase in cu_2oseo_3 . *Phys. Rev. Lett.*, 108:237204, Jun 2012. doi: 10.1103/PhysRevLett.108.237204. URL <http://link.aps.org/doi/10.1103/PhysRevLett.108.237204>.

- [18] H. Wilhelm, M. Baenitz, M. Schmidt, U. K. Rößler, A. A. Leonov, and A. N. Bogdanov. Precursor phenomena at the magnetic ordering of the cubic helimagnet fege. *Phys. Rev. Lett.*, 107:127203, Sep 2011. doi: 10.1103/PhysRevLett.107.127203. URL <http://link.aps.org/doi/10.1103/PhysRevLett.107.127203>.
- [19] X. Yu, A Kikkawa, D. Morikawa, K Shibata, Y. Tokunaga, Y. Taguchi, and Y. Tokura. Variation of skyrmion forms and their stability in mnsi thin plates. *Phys. Rev. B* 91, 054411, 2015.
- [20] Albert Fert, Vincent Cros, and Joao Sampaio. Skyrmions on the track. *Nat nano*, 8(3):152–156, Mar 2013. doi: 10.1038/nnano.2013.29. URL <http://dx.doi.org/10.1038/nnano.2013.29>. Print.
- [21] R. Tomasello, E. Martinez, R. Zivieri, L. Torres, M. Carpentieri, and G. Finocchio. A strategy for the design of skyrmion racetrack memories. *Scientific Reports*, 4(6784), 10 2014. doi: 10.1038/srep06784. URL <http://dx.doi.org/10.1038/srep06784>. Online.
- [22] *Shape-dependence of the barrier for skyrmions on a two-lane racetrack*, Sept. 2016. Spintronics IX. doi: doi:10.1117/12.2237624. URL <http://dx.doi.org/10.1117/12.2237624>. Proc. SPIE 9931.
- [23] Theodore L. Monchesky. Skyrmions: Detection with unpolarized currents. *Nat Nano*, 10(12):1008–1009, Dez 2015. doi: 10.1038/nnano.2015.226. URL <http://dx.doi.org/10.1038/nnano.2015.226>. News and Views.
- [24] U. K. Rößler, A. N. Bogdanov, and C. Pleiderer. Spontaneous skyrmion ground states in magnetic metals. *Nature* 442, 797-801, 2006.
- [25] F. N. Rybakov, A. B. Borisov, and A. N. Bogdanov. Three-dimensional skyrmion states in thin films of cubic helimagnets. *Phys. Rev. B* 87, 094424, 2013. doi: 10.1103/PhysRevB.87.094424. URL <http://journals.aps.org/prb/pdf/10.1103/PhysRevB.87.094424>.
- [26] A. Leonov. *Twisted, localized, and modulated states described in the phenomenological theory of chiral and nanoscale ferromagnets*. PhD thesis, Technische Universität Dresden, 2011. URL http://www.ifw-dresden.de/userfiles/groups/itf_folder/Theeses/andriy_leonov_phd.pdf.
- [27] N. S. Kiselev, A. N. Bogdanov, R. Schäfer, and U. K. Rössler. Chiral skyrmions in thin magnetic films: new objects for magnetic storage technologies? *J. Phys. D: Appl. Phys.* 44, 392001, 2011.

- [28] Gen Yin, Yizhou Liu, Yafis Barlas, Jiadong Zang, and Roger K. Lake. Topological spin hall effect resulting from magnetic skyrmions. *Phys. Rev. B*, 92:024411, Jul 2015. doi: 10.1103/PhysRevB.92.024411. URL <http://link.aps.org/doi/10.1103/PhysRevB.92.024411>.
- [29] Keita Hamamoto, Motohiko Ezawa, and Naoto Nagaosa. Quantized topological hall effect in skyrmion crystal. *Phys. Rev. B*, 92:115417, Sep 2015. doi: 10.1103/PhysRevB.92.115417. URL <http://link.aps.org/doi/10.1103/PhysRevB.92.115417>.
- [30] Xichao Zhang, Yan Zhou, and Motohiko Ezawa. Magnetic bilayer-skyrmions without skyrmion hall effect. *Nature Communications*, 7(10293), Jan 2016. doi: 10.1038/ncomms10293. URL <http://dx.doi.org/10.1038/ncomms10293>.
- [31] C. Reichhardt and C. J. Olson Reichhardt. Drive Dependence of the Skyrmion Hall Effect in Disordered Systems. *ArXiv e-prints*, May 2016.
- [32] Wanjun Jiang, Xichao Zhang, Guoqiang Yu, Wei Zhang, Xiao Wang, M. Benjamin Jungfleisch, John E. Pearson, Xuemei Cheng, Olle Heinonen, Kang L. Wang, Yan Zhou, Axel Hoffmann, and Suzanne G. E. te Velthuis. Direct observation of the skyrmion hall effect. *Nat Phys*, Sep 2016. doi: 10.1038/nphys3883. URL <http://dx.doi.org/10.1038/nphys3883>. advance online publication.
- [33] K. Litzius, I. Lemes, B. Krüger, P. Bassirian, L. Caretta, K. Richter, F. Büttner, K. Sato, O. A. Tretiakov, J. Förster, R. M. Reeve, M. Weigand, I. Bykova, H. Stoll, G. Schütz, G. S. D. Beach, and M. Kläui. Skyrmion Hall Effect Revealed by Direct Time-Resolved X-Ray Microscopy. *ArXiv e-prints*, August 2016.
- [34] Stavros Komineas and Nikos Papanicolaou. Skyrmion dynamics in chiral ferromagnets. *Phys. Rev. B*, 92:064412, Aug 2015. doi: 10.1103/PhysRevB.92.064412. URL <http://link.aps.org/doi/10.1103/PhysRevB.92.064412>.
- [35] D. Ehlers, I. Stasinopoulos, V. Tsurkan, H.-A. Krug von Nidda, T. Fehér, A. Leonov, I. Kézsmárki, D. Grundler, and A. Loidl. Skyrmion dynamics under uniaxial anisotropy. *Phys. Rev. B*, 94:014406, Jul 2016. doi: 10.1103/PhysRevB.94.014406. URL <http://link.aps.org/doi/10.1103/PhysRevB.94.014406>.
- [36] Junichi Iwasaki, Masahito Mochizuki, and Naoto Nagaosa. Current-induced skyrmion dynamics in constricted geometries. *Nat Nano*, 8(10):742–747, Oct. 2013. doi: 10.1038/nnano.2013.176. URL <http://dx.doi.org/10.1038/nnano.2013.176>.

- [37] J. Mohanty, B. Pfau, S. Schaffert, A. Bisig, M. Foerster, T. Schulz, C. A. F. Vaz, J. H. Franken, H. J. M. Swagten, M. Klaui, and S. Eisebitt. Dynamics and inertia of skyrmionic spin structures. *Nat Phys*, 11(3):225–228, Mar 2015. doi: 10.1038/nphys3234. URL <http://dx.doi.org/10.1038/nphys3234>.
- [38] J C Martinez and M B A Jalil. Topological dynamics and current-induced motion in a skyrmion lattice. *New Journal of Physics*, 18(3):033008, 2016. URL <http://stacks.iop.org/1367-2630/18/i=3/a=033008>.
- [39] Spirit atomistic spin code. <https://spirit-code.github.io/>, 2016. Accessed: 2016-10-25.
- [40] juspinx. <http://www.judft.de/pm/index.php?n=Codes.JuSpinX>, 2016. Accessed: 2016-11-07.
- [41] J H Mentink, M V Tretyakov, A Fasolino, M I Katsnelson, and Th Rasing. Stable and fast semi-implicit integration of the stochastic landau-lifshitz equation. *Journal of Physics: Condensed Matter*, 22(17):176001, 2010. URL <http://stacks.iop.org/0953-8984/22/i=17/a=176001>.
- [42] G.P. Müller. Exploration of skyrmion energy landscapes. Master thesis, RWTH Aachen and FZ Jülich, 2015. URL <http://www.fz-juelich.de/SharedDocs/Downloads/PGI/PGI-1/EN/Mueller-Gideon-Master.pdf>.
- [43] I. Dzyaloshinsky. A thermodynamic theory of "weak" ferromagnetism of antiferromagnetics. *Journal of Physics and Chemistry of Solids*, 4(4):241 – 255, 1958. ISSN 0022-3697. doi: [http://dx.doi.org/10.1016/0022-3697\(58\)90076-3](http://dx.doi.org/10.1016/0022-3697(58)90076-3). URL <http://www.sciencedirect.com/science/article/pii/0022369758900763>.
- [44] Tôru Moriya. Anisotropic superexchange interaction and weak ferromagnetism. *Phys. Rev.*, 120:91–98, Oct 1960. doi: 10.1103/PhysRev.120.91. URL <http://link.aps.org/doi/10.1103/PhysRev.120.91>.
- [45] Ashis Kumar Nandy, Nikolai S. Kiselev, and Stefan Blügel. Interlayer exchange coupling: A general scheme turning chiral magnets into magnetic multilayers carrying atomic-scale skyrmions. *Phys. Rev. Lett.*, 116:177202, Apr 2016. doi: 10.1103/PhysRevLett.116.177202. URL <http://link.aps.org/doi/10.1103/PhysRevLett.116.177202>. Supplemental Material.
- [46] Joseph Barker and Roy W. Chantrell. Higher-order exchange interactions leading to metamagnetism in ferh. *Phys. Rev. B*,

- 92:094402, Sep 2015. doi: 10.1103/PhysRevB.92.094402. URL <http://link.aps.org/doi/10.1103/PhysRevB.92.094402>.
- [47] L. Landau and E. Lifschitz. On the theory of magnetic permeability in ferromagnetic bodies. *Phys. Z. Sowjetunion* 8, 153-169, 1935.
- [48] T. L. Gilbert. Armor research foundation project no. a059, supplementary report, may 1, 1956. *Phys. Rev.*, 100:1243, 1955. Never published. Only abstract and summary available.
- [49] T. L. Gilbert. A phenomenological theory of damping in ferromagnetic materials. *IEEE Transactions on Magnetics*, 40(6):3443–3449, Nov 2004. ISSN 0018-9464. doi: 10.1109/TMAG.2004.836740. URL <http://dx.doi.org/10.1109/TMAG.2004.836740>.
- [50] D. S. G. Bauer. Atomistic spin-dynamics in confined magnetic nano-structures. Diploma thesis, RWTH Aachen and FZ Jülich, 2008. URL http://www.fz-juelich.de/SharedDocs/Downloads/PGI/PGI-1/EN/Bauer.diploma_pdf.pdf.
- [51] T. L. Gilbert and J. M. Kelly. Anomalous rotational damping in ferromagnetic sheets. In *Conf. Magnetism and Magnetic Materials, Pittsburgh, PA, June 14-16, 1955*, pages 253–263. NewYork: American Institute of Electrical Engineers, Oct 1955.
- [52] J.C. Slonczewski. Current-driven excitation of magnetic multilayers. *Journal of Magnetism and Magnetic Materials*, 159(1-2):L1 – L7, 1996. ISSN 0304-8853. doi: [http://dx.doi.org/10.1016/0304-8853\(96\)00062-5](http://dx.doi.org/10.1016/0304-8853(96)00062-5). URL <http://www.sciencedirect.com/science/article/pii/0304885396000625>.
- [53] J.C Slonczewski. Excitation of spin waves by an electric current. *Journal of Magnetism and Magnetic Materials*, 195(2):L261 – L268, 1999. ISSN 0304-8853. doi: [http://dx.doi.org/10.1016/S0304-8853\(99\)00043-8](http://dx.doi.org/10.1016/S0304-8853(99)00043-8). URL <http://www.sciencedirect.com/science/article/pii/S0304885399000438>.
- [54] L. Berger. Emission of spin waves by a magnetic multilayer traversed by a current. *Phys. Rev. B*, 54:9353–9358, Oct 1996. doi: 10.1103/PhysRevB.54.9353. URL <http://link.aps.org/doi/10.1103/PhysRevB.54.9353>.
- [55] L. Berger. Interaction of electrons with spin waves in the bulk and in multilayers (invited). *Journal of Applied Physics*, 91(10):6795–6800, 2002. doi: <http://dx.doi.org/10.1063/1.1446121>. URL <http://scitation.aip.org/content/aip/journal/jap/91/10/10.1063/1.1446121>.

- [56] E. B. Myers, D. C. Ralph, J. A. Katine, R. N. Louie, and R. A. Buhrman. Current-induced switching of domains in magnetic multilayer devices. *Science*, 285(5429):867–870, 1999. ISSN 0036-8075. doi: 10.1126/science.285.5429.867. URL <http://science.scienmag.org/content/285/5429/867>.
- [57] J. A. Katine, F. J. Albert, R. A. Buhrman, E. B. Myers, and D. C. Ralph. Current-driven magnetization reversal and spin-wave excitations in co /cu /co pillars. *Phys. Rev. Lett.*, 84:3149–3152, Apr 2000. doi: 10.1103/PhysRevLett.84.3149. URL <http://link.aps.org/doi/10.1103/PhysRevLett.84.3149>.
- [58] J. Grollier, V. Cros, A. Hamzic, J. M. George, H. Jaffres, A. Fert, G. Faini, J. Ben Youssef, and H. Legall. Spin-polarized current induced switching in co/cu/co pillars. *Applied Physics Letters*, 78(23):3663–3665, 2001. doi: <http://dx.doi.org/10.1063/1.1374230>. URL <http://scitation.aip.org/content/aip/journal/apl/78/23/10.1063/1.1374230>.
- [59] Z. Li and S. Zhang. Magnetization dynamics with a spin-transfer torque. *Phys. Rev. B*, 68:024404, Jul 2003. doi: 10.1103/PhysRevB.68.024404. URL <http://link.aps.org/doi/10.1103/PhysRevB.68.024404>.
- [60] P. Chureemart, R. F. L. Evans, and R. W. Chantrell. Dynamics of domain wall driven by spin-transfer torque. *Phys. Rev. B*, 83:184416, May 2011. doi: 10.1103/PhysRevB.83.184416. URL <http://link.aps.org/doi/10.1103/PhysRevB.83.184416>.
- [61] Leonhard Euler. *Institutiones calculi differentialis*. Lipsiae Et Berolini, 1755.
- [62] B Skubic, J Hellsvik, L Nordström, and O Eriksson. A method for atomistic spin dynamics simulations: implementation and examples. *Journal of Physics: Condensed Matter*, 20(31):315203, 2008. URL <http://stacks.iop.org/0953-8984/20/i=31/a=315203>.
- [63] D. Stauffer. *Annual Reviews of Computational Physics: IX*. Annual Reviews of Computational Physics. World Scientific, 2001. ISBN 9789812811578. URL <https://books.google.de/books?id=Zd-YQEzJ008C>.
- [64] Qt application framework. <http://doc.qt.io/>, 2016. Accessed: 2016-10-19.
- [65] Vtk. <http://www.vtk.org/>, 2016. Accessed: 2016-11-07.
- [66] emscripten. <http://kripken.github.io/emscripten-site/>, 2016. Accessed: 2016-10-25.

BIBLIOGRAPHY

- [67] Filipp N. Rybakov, Aleksandr B. Borisov, Stefan Blügel, and Nikolai S. Kiselev. New type of stable particlelike states in chiral magnets. *Phys. Rev. Lett.*, 115:117201, Sep 2015. doi: 10.1103/PhysRevLett.115.117201. URL <http://link.aps.org/doi/10.1103/PhysRevLett.115.117201>.

DECLARATION

I hereby declare that this thesis is my original work and it has been written by me in its entirety. I have acknowledged all the sources of information which have been used in the thesis. This thesis has not been submitted, in whole or in part, to any other examination authority for any other degree.

Forschungszentrum Jülich, November 2016

Daniel Christoph Beppo
Schürhoff