Advanced methods for atomic scale spin simulations and application to localized magnetic states

Gideon Philipp Müller

Schlüsseltechnologien / Key Technologies
Band / Volume 205
ISBN 978-3-95806-432-4

Mitglied der Helmholtz-Gemeinschaft

JÜLICH
Forschungszentrum

Schriften des Forschungszentrums Jülich
Reihe Schlüsseltechnologien / Key Technologies Band / Volume 205

Forschungszentrum Jülich GmbH
Peter Grünberg Institut (PGI)
Quanten-Theorie der Materialien (PGI-1/IAS-1)

Advanced methods for atomic scale spin simulations and application to localized magnetic states

Gideon Philipp Müller

Schriften des Forschungszentrums Jülich
Reihe Schlüsseltechnologien / Key Technologies Band / Volume 205

ISSN 1866-1807 ISBN 978-3-95806-432-4

Bibliografische Information der Deutschen Nationalbibliothek.
Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der
Deutschen Nationalbibliografie; detaillierte Bibliografische Daten
sind im Internet über http://dnb.d-nb.de abrufbar.

Herausgeber
und Vertrieb: Forschungszentrum Jülich GmbH
Zentralbibliothek, Verlag
52425 Jülich
Tel.: +49 2461 61-5368
Fax: +49 2461 61-6103
zb-publikation@fz-juelich.de
www.fz-juelich.de/zb

Umschlaggestaltung: Grafische Medien, Forschungszentrum Jülich GmbH

Druck: Grafische Medien, Forschungszentrum Jülich GmbH

Copyright: Forschungszentrum Jülich 2019

Schriften des Forschungszentrums Jülich
Reihe Schlüsseltechnologien / Key Technologies, Band / Volume 205

D 82 (Diss. RWTH Aachen University, 2019)

ISSN 1866-1807
ISBN 978-3-95806-432-4

Vollständig frei verfügbar über das Publikationsportal des Forschungszentrums Jülich (JuSER)
unter www.fz-juelich.de/zb/openaccess.

This is an Open Access publication distributed under the terms of the Creative Commons Attribution License 4.0,
which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

FOREWORD

A few short words on progress. I believe that the truth of scientific progress and understanding lies in simplification in its various forms – in the sense of distillation and reduction to leave only that which is essential. It might mean that a new generation has an intuition about ideas too abstract for the previous generation, a prime example of which is how quantum mechanics is now taught to Physics students already in their second year. Another facet might be a new mathematical representation in which an idea suddenly becomes clearer. On the other hand one might have to take a step back and revisit old ideas and representations and reformulate them.

This is the progress I set out to achieve in my work: simplifying and improving the mathematical and computational tools for the methods in this field of science for the next generation of students, so that they may find them more obvious, clearer and easier to use, in one word simpler.

Simplification can also mean easier access to knowledge, which is why significant stress was placed on the open availability of the written software and it was designed to be easy to use by anyone, anywhere and on any device. My wish for this project was not simply to have people use the software, but to enhance collaboration by helping the exchange of results and reducing the barrier to view and even reproduce the work of others.

Many of the goals I set for my project I have achieved and I firmly believe that the directions taken are the right ones and that the tools which have been created will make it easier for new scientists in magnetism to gain an intuitive understanding of the systems they study and the methods they apply. Furthermore, I maintain that the mathematical framework presented here has the potential to describe more methods and models from the same point of view.

ACKNOWLEDGEMENTS

First of all I would like to thank Prof. Stefan Blügel and Prof. Hannes Jónsson for providing me with the great opportunity of writing my Dissertation in a joint collaboration between the Forschungszentrum Jülich and the University of Iceland. Together with the various conferences I was allowed to attend, this has provided me with many chances to broaden my horizon and learning new languages by spending time abroad and collaborating across many borders. Furthermore, their general guidance and perspectives were a great help in making this thesis successful.

I would like to express my very great appreciation to Dr. Nikolai Kiselev for valuable and constructive discussions, thorough feedback and helpful suggestions throughout the entire course of this research work. His broad knowledge of the subject field have provided practical context and helpful insight and placed this work at the cutting edge.

My special thanks go to Dr. Pavel Bessarab for introducing me to important, interesting topics of this work, guiding me towards more abstract methodology and increasing my interest in the field and to Dr. Filipp Rybakov for guiding several of the software implementation and user interface design decisions.

I would also like to extend my thanks to Markus Hoffmann, Daniel Schürhoff, Stefanos Mavros and Moritz Sallermann for their collaborations and the help in creating and testing the code developed for this thesis, Dr. Sergei Vlasov and Fabian Lux for their collaborations and helping me better understand and describe the mathematical framework, as well as Stephan von Malottki, Jan Masell and Mariia Filianina for various helpful discussions.

Thanks also to Florian Rhiem for programming the tremendously useful visualisation library – used heavily for visualisations throughout this thesis – as well as the Web UI, and for help with general programming and IT-related problems.

Finally I wish to thank my family and my girlfriend Mina for their encouragement and support throughout this work.

ABSTRACT

An active field of research in magnetism today involves studies of solitons – localised magnetic textures possessing particle-like properties. They are considered promising for various applications but are also intriguing from a fundamental point of view. Most of the effects related to magnetic solitons, including in particular skyrmions, can be described in classical spin-lattice models. In this context, effective tools for materials and device design are needed in order to calculate properties, such as thermal stability, lifetime, critical velocity, characteristic dynamical modes and much more. This thesis is devoted to the development of new methodology and the implementation and verification of a new software framework for the simulation of atomistic spin systems.

Going beyond the widely known approaches of Monte Carlo and Landau-Lifshitz-Gilbert (LLG) dynamics, this thesis describes the recently developed geodesic nudged elastic band (GNEB) method and harmonic transition state theory in a consistent mathematical framework. The minimum mode following (MMF) method, which can be used to seek out first order saddle points in the energy landscape, is formulated for magnetic systems. Such saddle point searches are an essential part in identifying possible transition processes between magnetic configurations and therefore in estimating the rates of transitions between magnetic states, which determine the states' lifetimes. Using the MMF method, a mitosis-like skyrmion duplication – or inversely a merger – transition was found and could be reproduced in LLG dynamics simulations using an external magnetic field pulse.

The entire set of methods discussed in this thesis has been implemented into a novel, open source software framework. Using scripting and graphical user interfaces, including powerful real-time visualisation features, the methods can now be used easily in conjunction with and complementary to one another. The implementation, including high performance parallelisation schemes, is described and a key set of its features are demonstrated.

The software framework is applied to a variety of challenging problems in two- and three-dimensional systems. In two dimensions, complex higher-order skyrmionic textures are studied using the GNEB method and mitosis-like transitions identified. Three-dimensional systems are shown to host a large variety of complex spin textures, including a novel three-dimensionally localised state – the magnetic globule. This state is composed of two coupled quasi-monopoles, also known as Bloch points, and may form stable spin textures in a wide range of parameters and in various situations.

The software framework presented here brings simulations of atomic scale magnetic systems to a higher level and represents a significant step in the modernisation of computational tools in magnetism. It brings benefits in productivity and ease of use and improves accessibility of recent and novel methodology.

KURZZUSAMMENFASSUNG

Ein zur Zeit aktives Forschungsfeld im Bereich des Magnetismus bezieht sich auf sogenannte Solitonen – lokalisierte magnetische Texturen mit teilchenartigen Eigenschaften. Sie werden für diverse Anwendungen als vielversprechend angesehen, sind aber auch von einem fundamentalen Blickwinkel aus interessant. Die meisten Effekte im Zusammenhang mit magnetischen Solitonen, insbesondere Skyrmionen, können im Rahmen klassischer Spin-Gitter Modelle beschrieben werden. In diesem Kontext werden effektive Werkzeuge für das Entwerfen von Materialien und Anordnungen benötigt, um Eigenschaften wie thermische Stabilität, Lebenszeit, kritische Geschwindigeit, charakteristische dynamische Eigenmoden und viel mehr zu berechnen. Diese Arbeit widmet sich der Entwicklung neuer Methodik und dem Implementieren und Verifizieren einer neuen Software für die Simulation atomistischer Spin-Systeme.

Über die weithin bekannten Ansätze der Monte Carlo-Methode und Landau-Lifshitz-Gilbert (LLG) Dynamik hinaus werden die kürzlich entwickelte "geodesic nudged elastic band" (GNEB) Methode und die "harmonic transition state theory" in einem konsistenten mathematischen Rahmen beschrieben. Die "minimum mode following" (MMF) Methode, welche verwendet werden kann, um Sattelpunkte in der Energielandschaft ausfindig zu machen, wird für magnetische Systeme formuliert. Solche Sattelpunktsuchen spielen eine wichtige Rolle in der Identifikation möglicher Übergangsprozesse zwischen magnetischen Konfigurationen und somit in der Bestimmung der Übergangsraten, welche die Lebenszeiten magnetischer Zustände bestimmen. Mit der MMF-Methode wurde eine mitoseartige Skyrmionverdoppelung – invers betrachtet eine Kollision – gefunden und in LLG Dynamik-Simulationen mit Hilfe eines magnetischen Feld-Pulses reproduziert.

Die gesamte Menge der Methoden, welche in dieser Arbeit diskutiert werden, wurde in einer neuen, frei verfügbaren Software implementiert. Mithilfe von Skripten und grafischen User-Interfaces, einschließlich Echtzeitvisualisierung, können die Methoden nun leicht zusammen und komplementär zu einander verwendet werden. Die Implementierung wird einschließlich hochperformanter Parallelisierungen beschrieben und die wichtigsten Merkmale werden demonstriert.

Die Software wird auf eine Vielzahl herausfordernder Probleme in zwei- und dreidimensionalen Systemen angewandt. In zwei Dimensionen werden komplexe skyrmionische Texturen höherer Ordnung unter Verwendung der GNEB-Methode studiert und weitere mitoseartige Übergänge identifiziert. Es wird gezeigt, dass dreidimensionale Systeme eine große Vielzahl komplexer Spin-Texturen ermöglichen, einschließlich eines neu entdecken dreidimensional lokalisierten Zustandes – dem "magnetic globule". Dieser Zustand besteht aus zwei aneinander gekoppelten Quasimonopolen, auch Bloch-Punkte genannt, und kann in einem großen Parameterbereich und in diversen Situationen stabile Spin-Texturen formen.

Die hier vorgestellte Software bringt Simulationen von magnetischen Systemen auf der atomistischen Skala auf ein höheres Niveau und stellt einen nennenswerten Schritt in der Modernisierung der numerischen Werkzeuge im Magnetismus dar. Sie bringt Vorteile in Produktivität und einfacher Benutzung und verbessert die Zugänglichkeit kürzlich entwickelter, sowie neuer Methodik.

ÁGRIP

Staðbundin segulástönd, svokallaðir einfarar, hafa agnaeiginleika og eru mikið rannsökuð í dag. Vonir standa til að hægt verði að nota þau á ýmsan hátt í framtíðar tækni en grundvallar eiginleikar þeirra eru einnig mjög áhugaverðir. Að mestu leiti er hægt að lýsa slíkum ástöndum með klassískum spuna-grindar líkönum. Mikilvægt er að hafa skilvirkar aðferðir til að reikna út hina ýmsu eiginleika svo sem varmafræðilegan stöðugleika, líftíma, krítískan hraða, eiginhætti o.sv.frv. Þessi ritgerð fjallar um þróun aðferðafræði, innsetningu og beitingu nýs hugbúnaðar fyrir tölvureikninga á spunakerfum á atómskala.

Til að komast umfram það sem hægt er að gera með algengum aðferðum svo sem Monte Carlo og Landau-Lifshitz-Gilbert (LLG) tímaferlum, byggir ritgerðin á heilsteyptri framsetningu á 'geodesic nudged elastic band' (GNEB) aðferðinni og kjörsveifilsnálgun á virkjunarástandskenningunni (HTST). 'Minimum mode following' (MMF) aðferðin til að finna fyrsta stigs söðulpunkta er útvíkkuð fyrir segulkerfi. Slíkar söðulpunktaleitir eru mikilvægar til að finna mögulega hvarfganga fyrir breytingar úr einu segulástandi í annað sem er nauðsynlegur undanfari þess að meta hvarfhraða og þar með líftíma segulástanda. Með MMF aðferðinni var fundinn hvarfgangur þar sem skyrmeind skiptist upp í tvær skyrmeindir, eða öfugt – að tvær skyrmeindir renna saman í eina – og sýnt fram á að hægt væri að kalla fram slíkan hvarfgang í LLG reikningum á tímaframvindu þegar beitt er segulsviðspúlsi.

Hinar ýmsu aðferðir sem fjallað er um í ritgerðinni hafa verið innleiddar í nýjum hugbúnaðarpakka. Með skriftum og sjónrænu viðmóti sem og öflugri rauntíma myndgerfingu er auðvelt að bbeita hinum ýmsu aðferðum og nýta kosti hverrar um sig. Innsetningu aðferðanna er lýst sem og lykil eiginleikum hugbúnaðarins.

Hugbúnaðinum er beitt á ýmis krefjandi verkefni sem tengjast tví- og þrívíðum kerfum. Í tvívíðum kerfum eru skyrmeindir rannsakaðar með GNEB aðferðinni og hún notuð til að greina skiptingu skyrmeindar. Fyrir þrívíð kerfi er sýnt fram á að ýmis flókin segulkerfi geta myndast, þar með nýtt þrívítt segulkerfi sem nefnt hhefur verið 'globule'. Þetta ástand felur í sér tvö tengd einskaut, þ.e. Bloch punkta, og það getur verið stöðugt fyrir mörg stikugildi og við ýmsar aðstæður. Hubúnaðurinn gerir atómskala reikninga á segulkerfum auðvelda og felur í sér stórt skref til að nútímavæða þau tól sem hægt er að nota í tölvureikningum á segulkerfum. Hann sameinar kosti þess að vera auðveldur í notkun og eykur aðgengi að nýrri- og endurbættri aðferðafræði.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

CODE LISTINGS

ACRONYMS

MD	magnetic domain
DW	domain wall
HDD	hard disk drive
SSD	solid state drive
DFT	density functional theory
ASD	atomistic spin dynamics
MC	Monte Carlo
LLG	Landau-Lifshitz-Gilbert
DMI	Dzyaloshinskii-Moriya interaction
FFT	fast Fourier transform

MEP	minimum energy path
NEB	nudged elastic band
GNEB	geodesic nudged elastic band
MMF	minimum mode following
TST	transition state theory
HTST	harmonic transition state theory
CI	climbing image
BP	Bloch point
SP	saddle point
MG	magnetic globule
API	application programming interface
GUI	graphical user interface
CPU	central processing unit
GPU	graphics processing unit
MPI	message passing interface

INTRODUCTION

I.1 GENERAL

Magnetism is a large and vibrant field of Physics with a long history and was in fact discovered in ancient times, when humans observed the unusual behaviour of lodestones. Since the invention of the compass, countless further applications have been invented and magnetism has become ubiquitous in our daily lives. One of the most evident examples of its application which is broadly used today, is the hard disk drive (HDD), which has played a major role in the development of modern computer technology. The key concept of the technology, the giant magnetoresistance (GMR) effect [1–3], led to a Nobel prize for P. Grünberg and A. Fert in 2007. The HDD has driven miniaturisation of data storage and due to the durability of the read/write processes and the long lifetime of the information, it has become a key component in datacentres. Mechanical components, however, limit the speed of the reading and writing processes and also the thermal efficiency. In contrast to flash storage – used in solid state drives (SSDs), portable data storage such as USB flash drives or mobile devices – HDDs store data on magnetic material in the form of magnetic domains pointing in one of two directions. SSDs use integrated electric circuits for persistent data storage and thereby avoid mechanically moving parts and increase reading and writing speed, but suffer from shorter lifetimes.

A magnetic racetrack memory design was developed aiming to alleviate these limitations [4, 5]. By applying electric currents, as illustrated in Figure I.1, domain walls can be moved along a magnetic strip at high speed. Analogous to the HDD, the binary information is encoded in the form of domains, but now stationary read/write heads can be used and no mechanical parts are needed as the domains themselves are movable. This development has become a powerful impetus in the field of non-collinear magnetism.

The domain wall racetrack design, however, can only be competitive when manufacturing costs of racetracks are sufficiently low. At the same time, the material must provide a sufficiently low depinning current for the movement of the domain walls and their speed of movement at such currents must be sufficiently high. Finally, the track needs to be miniaturised enough for application in modern devices. Especially for the usage in mobile devices like mobile phones, tablets and laptops, energy efficiency and miniaturisation are fundamental. Note that cost factors also play an important role in device design. In spite of rapid developments, the cost of storing a

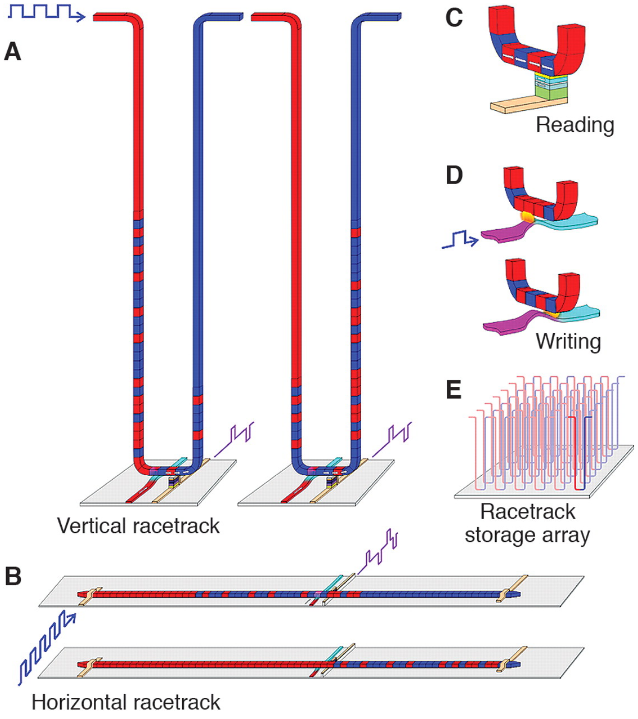
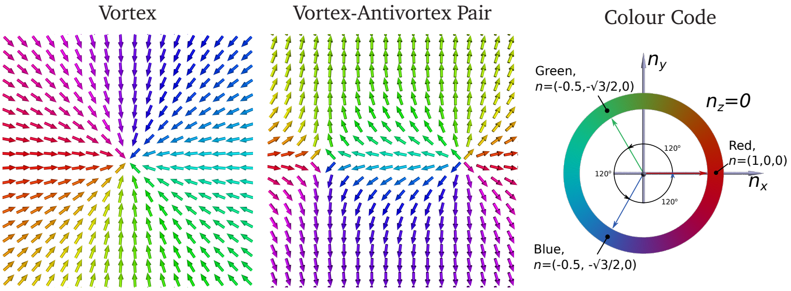


FIGURE I.1 – Domain wall racetrack memory concept

The racetrack is a ferromagnetic nanowire, where the data is encoded as a pattern of magnetic domains. Spin-polarized currents move the entire pattern of domain walls (DWs) past stationary read and write elements. (A) A vertical-configuration racetrack could increase storage density and the magnetic patterns before and after the movement of the DWs past the read and write elements. (B) A horizontal, flat configuration. (C) Reading data from the stored pattern is done by measuring the magnetoresistance of a magnetic tunnel junction element connected to the racetrack. (D) Writing data can be accomplished by the fringing fields of a DW moved in a second nanowire placed at a right angle to the storage nanowire. (E) Illustration of an array of racetracks, built on a chip to enable high-density storage. From [4]. Reprinted with permission from AAAS.

certain amount of data on an HDD remains approximately 100 times cheaper than in a (random access) SSD. This underlines the potential benefits of a data storage device that combines the low cost of an HDD with the high performance of an SSD.

Subsequently, interesting theoretical predictions from condensed matter physics sparked interest in topology and the emergence of vortex-like structures. One of the more recent but also most prominent examples is the 2016 Nobel prize which was awarded to D. J. Thouless, F. D. M. Haldane and J. M. Kosterlitz for their work on topological phases of matter and phase transitions [6]. Said topological phase transitions are driven by vortices and anti-vortices, representing topological defects, as shown in Figure I.2.


Vortex

Vortex-Antivortex Pair

Colour Code

FIGURE I.2 – Topological vortices in 2D and colour code visualisation

A vortex and vortex-antivortex pair in the XY-Heisenberg model and the corresponding hue-lightness (HL) colour code, used throughout this thesis, are shown. The in-plane component is encoded by the hue, while the z-component is encoded by the lightness of the colour, white being $n_z = 1$ and black being $n_z = -1$. The vortex images, captured in the graphical user interface (GUI) of *Spirit*, are analogous to the informational paper of the 2016 Nobel prize [6]. Colour code visualisation provided by Nikolai Kiselev.

In magnetism, analogies have been drawn to the vortex-like magnetic skyrmions, such as shown in Figure I.3, which also carry a topological charge. These highly noncollinear structures were predicted to exist in magnetic materials [7–14] and are generally stabilised by antisymmetric exchange interaction (usually called Dzyaloshinskyi-Moriya interaction) [15, 16] or frustrated exchange interaction. Their existence was later verified experimentally [17–20].

Due to their nontrivial topology, skyrmions possess interesting transport properties and can, for example, be propagated by electric currents [21–24] – much like domain walls – and are therefore considered interesting and promising for various applications, as well as for fundamental research. In terms of application, they have attracted significant attention due to their stability and compact size down to a few nano-meters [25] or even a few atomic distances [26]. This characteristic makes them promising candidates for a novel concept of racetrack memory [27–29], in which domain walls as data carriers are replaced by skyrmions. Compared to domain walls,

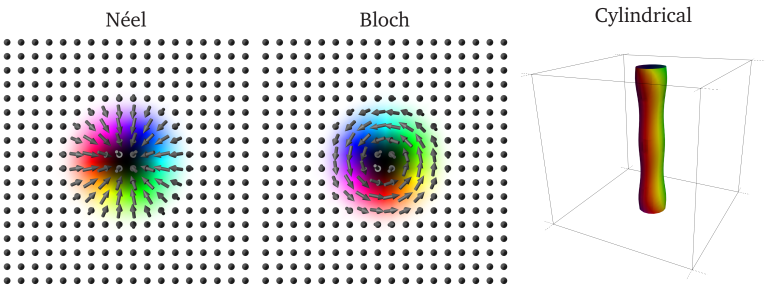


FIGURE I.3 – Skyrmions in two and three dimensions

A Néel skyrmion and a Bloch skyrmion in a monolayer, as well as the $n_z = 0$ isosurface of a cylindrical skyrmion tube in a three-dimensional thin layer. The modulations of the skyrmion tube along its axis appear due to a conical background and the free surfaces of the sample (see also Chapter 4).

they are more movable and less easily pinned by defects [21, 30, 31]. These two properties can facilitate the aim of reducing the currents required to drive their motion.

An essential part of spintronic device design is the theoretical prediction of material and device properties, from analytical results [7, 32–34] or numerical calculations [35–38]. The ability to predict experimental outcome by calculation can significantly decrease the effort required to design novel material compositions and systems. It can even point in new directions of research, as it is often easier to explore a wide parameter space by computation than by experiment.

For numerical calculations, it is common to employ multiscale approaches [39], as physical parameters should ideally be derived from first principles calculations, such as density functional theory (DFT), in order to understand the physics from a fundamental point of view. As the numerical cost of DFT and other *ab initio* methods is comparatively high and it is desirable to generate predictions over a broad range of length and time scales, their output is transformed into model parameters on a higher level.[1]

The skyrmion racetrack, as shown in Figure I.4, is an example where such a multiscale approach can be successfully applied [38, 40], for instance in order to guide designs where skyrmions are most stable while being easily created when needed. While this work is embedded in the currently highly active field of skyrmionics and the search for next-generation data devices [41], nanomagnetism in general is a wide field with a large variety of other potential applications [42].

Assuming a continuum field of the magnetisation distribution in a system, both the aforementioned domain wall and skyrmion textures are topological solitons, i.e. solutions to a system of partial differential equations homotopically distinct from the

[1] Parameters can also be heuristically modelled after experimental data in order to achieve good agreement.

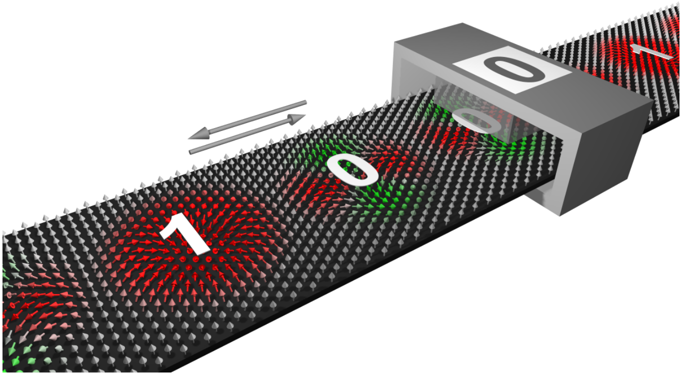


FIGURE I.4 – Skyrmions and antiskyrmions on a racetrack

Schematic of a skyrmion-antiskyrmion racetrack device, where an electric current drives the skyrmions to move past a read/write head. As illustrated, data bits "0" and "1" in this device would be encoded by skyrmions and antiskyrmions respectively. Image provided by Markus Hoffmann.

trivial solution. As will be illustrated later in this work, chiral magnetic systems may host a wide variety of such solitons, even in two dimensions. In three-dimensional systems, the complexity is drastically increased and additional states, such as the Hopfion, become relevant [43].

As developments, such as the skyrmion racetrack, represent a move towards three-dimensional microelectronic devices and nanomagnetism, it is crucial to understand the complex magnetic ordering which can take place inside a 3D crystal. While skyrmions are already under considerable investigation also in 3D systems [13], Hopfions and other more complicated solitons are only now gaining attention [44–46]. Three-dimensional solitons may turn out to be crucial for the next level of data storage designs, moving away from planar arrangements.

This thesis provides the necessary tools for studying a very broad variety out of the effects mentioned above. By using the *Spirit* framework [47, 48], magnetic states and potential applications can be examined not only quantitatively, but also intuitively and qualitatively. *Spirit* is an open source,[2] publicly available software framework for spin simulations, designed and developed mainly by the author as part of this work.

There are a number of well-known tools for micromagnetism, which have been rigorously tested and provide powerful features, two prominent examples being `mu-Max` [49] and `OOMMF` [50]. These software tools have certain GUI elements, which simplify some processes and help with post processing. However, many parameters cannot be changed during a calculation and the visualisation of complex vector fields is limited. The same applies to atomistic codes, such as `UppASD` [51] and `VAMPIRE` [52],

[2] spirit-code.github.io

which for the most part have no GUIs at all and therefore require a significant level of knowledge before they can be used. An important goal behind the idea of *Spirit* was to create a tool which can be used "*out of the box*", i.e. which is intuitively clear in its input and output, and also provides advanced *in situ* controls: manual interaction with the vector field, control of all simulation parameters and complex initialisation of states. Another issue was the lack of non-Fortran implementations of methods beyond Monte Carlo (MC) or Landau-Lifshitz-Gilbert (LLG) dynamics. While Fortran is known as a powerful programming language, especially for numerical tasks, with many years of tradition in science, it produces many difficulties when creating a GUI, running the software on various different platforms and parallelising it on graphics processing units (GPUs). C++ has no disadvantages in terms of performance, but the aforementioned problems are easily solved by the abundance of freely available libraries.

For these reasons, the Fortran `JuSpinX` [53, 54] code was re-written as *Spirit* – a C++ software library together with a GUI and the additional goal to integrate CPU and GPU parallelisation.

Besides creating a GUI, a major goal of this software development is the unification of various computational methods commonly applied to atomistic (and in part also to micromagnetic systems): Monte Carlo (MC) and Landau-Lifshitz-Gilbert (LLG) dynamics [55], geodesic nudged elastic band (GNEB) method [56], harmonic transition state theory (HTST) [57], minimum mode following (MMF) [58] and the visualisation of eigenmodes, see also Table I.1.

Derivative	Method
None	Monte Carlo
Gradient	Direct energy minimisation Landau-Lifshitz-Gilbert dynamics Minimum energy path calculations
Hessian	Transition rate calculations Saddle point searches and mode following Calculation of eigenspectra

TABLE I.1 – Methods organised by required derivatives

The different methods discussed in this work are organised by the order of derivative of the energy with respect to the spin orientation. The gradient is the vector of first order derivatives, while the Hessian is the matrix of second order derivatives.

All of these methods are quite distinct from one another and each one has various different applications [59]. However, they partly build up on each other. For example, LLG can be used to find local minima of the energy landscape, GNEB and/or MMF can be used to find first-order saddle points (corresponding to transition states for thermally assisted transitions), and finally HTST can be applied to estimate the lifetime of a metastable magnetic configuration.

This thesis deals largely with the development and extension of the *Spirit* software, developed mainly by the author during the course of this work, into which all of the above methods have been implemented. As this revolved around the variety of different methods in the field and their integration into a single multifunctional tool for atomistic spin systems, this thesis reviews some well-known methods and presents their verification against the literature. Such verifications and other aspects from this thesis have been published by the author in Ref. [47].[3] The thesis is organised in the following way:

In Section I.2, the classical atomistic spin model will be presented as a basis for the entire work. Then, the well-known Monte Carlo and Landau-Lifshitz-Gilbert dynamics methods will be shortly reviewed in Section I.3 and Section I.4 respectively. Their implementation in *Spirit* – performed mainly by the author, with some support from collaborators – will be verified against known results. It will be illustrated that these basic methods are insufficient for gaining a complete picture of a magnetic spin system and Chapter 1 will deal with the more recently developed geodesic nudged elastic band and harmonic transition state theory methods. These enable the statistical estimation of magnetic state lifetimes from fundamental properties of the energy landscape. The existing mathematical derivations will be reproduced in the extrinsic coordinate view chosen by the author and the implementation will be verified. Both methods will subsequently be applied to produce novel results, where the author was a collaborator on Ref. [38]. The need for an additional method for rate theory in spin systems will be laid out. in Chapter 2, the mathematical basis derived – with aid from collaborators – and the author's implementation into *Spirit* demonstrated by the calculation of novel results. The method and its application have been published in Ref. [58]. *Spirit* will be discussed from a software point of view in Chapter 3. This software as a whole was developed mainly by the author, but it should be noted that the visualisation library *VFRendering* used inside the GUI of *Spirit*, was created by Florian Rhiem [60] and that several contributions to specific features of *Spirit* were made by students under the supervision of the author.[4] The novelty of its various features and the potential productivity improvements will be demonstrated and the software design explained. This has been published by the author in Ref. [47].[5] In Chapter 4, *Spirit* will be applied to complex three-dimensional spin configurations. The features of *Spirit* and a third-party simulation software will be used to uncover details about Bloch points in the atomistic model and to discover a new, stable particle-like state

[3] The paper is on arxiv and currently under review at Physical Review B

[4] A detailed list of contributors and their contributions can be found in the repository [48]

[5] The paper is on arxiv and currently under review at Physical Review B

in chiral magnets. This has been published by the author in Ref. [61].[6]

I.2 MODEL

As mentioned in the previous Section I.1, *ab initio* DFT calculations can be used to determine parameters, such as the exchange interaction constants, of the extended atomistic Heisenberg model, which in turn can be averaged out into parameters of the micromagnetic model, such as spin stiffness. This work concerns itself with cases where the approximation of the classical atomistic Heisenberg Hamiltonian is valid (see also Figure I.5). While the methods presented in the following chapters are generally also applicable to the micromagnetic model, it has some limitations which may inhibit especially the use of stochastic and rate theory methods (see Chapter 1) due to the assumption of $T = 0$ K and the possible appearance of topological defects respectively.

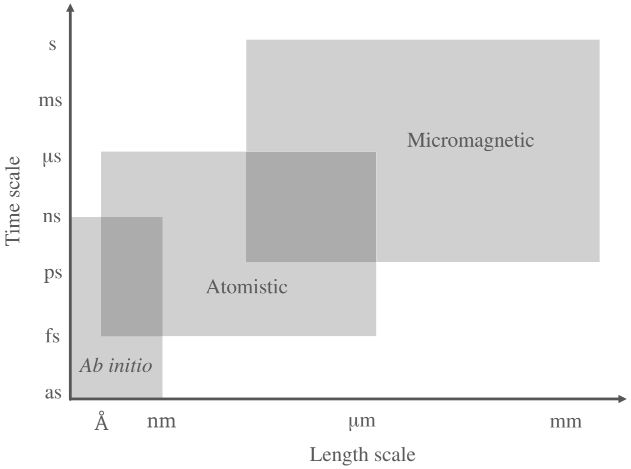


FIGURE I.5 – Hierarchy of time and length scales

The different scales of magnetisation dynamics covered by *ab initio*, atomistic and micromagnetic methods in terms of length, time, and frequency. The atomistic model bridges the gap between *ab initio* and micromagnetic simulations. Figure reprinted with minor modifications from [62] (see also [63]) with permission from IOP Publishing.

This section will explain the atomistic classical Heisenberg Hamiltonian, as well as its continuum limit, and then also derive the first and second order derivatives of the energy, i.e. the gradient vector and the Hessian matrix respectively. This will lay the

[6] The paper is on arxiv and currently under review at Physical Review Letters

foundation for the entire thesis, as these will be needed repeatedly, especially for the higher-order methods (see Table I.1).

I.2.1 *The extended Heisenberg Hamiltonian*

The relation of the quantum Heisenberg model and the classical model, where spins are represented by classical vectors $\vec{n}_i$ instead of operators, will not be discussed in detail here, as this has been done extensively in the literature. The derivations can be found in Refs. [64–68] and a historical overview and detailed description of this model can be found in Refs. [69, 70]. Due to the fact that the parameters of the atomistic model can be determined from electronic structure calculations, with results comparing well with experiment, this semi-classical approach has proven useful. Besides systems which cannot be approximated to the atomistic view, the major drawback of using a classical Heisenberg model is the low-temperature behaviour, where quantum effects can play a major role. Note that the modelling of high-temperature behaviour is also nontrivial.

In *Spirit*, the classical Heisenberg Hamiltonian is implemented in a general extended form

$$\begin{aligned}\mathcal{H} = &-\sum_{\langle ij\rangle} J_{ij}\vec{n}_i\cdot\vec{n}_j - \sum_{\langle ij\rangle}\vec{D}_{ij}\cdot(\vec{n}_i\times\vec{n}_j),\\ &-\sum_i \mu_i\vec{B}\cdot\vec{n}_i - \sum_i\sum_j K_j(\hat{K}_j\cdot\vec{n}_i)^2\\ &+\frac{1}{2}\frac{\mu_0}{4\pi}\sum_{\substack{i,j\\ i\neq j}}\mu_i\mu_j\frac{(\vec{n}_i\cdot\hat{r}_{ij})(\vec{n}_j\cdot\hat{r}_{ij})-\vec{n}_i\vec{n}_j}{{r_{ij}}^3}\,,\end{aligned}\tag{I.1}$$

where $\langle ij\rangle$ denotes unique pairs of interacting spins i and j, J_{ij} are parameters of the symmetric exchange (Heisenberg exchange interaction), $\vec{D}_{ij}$ are the parameters of the antisymmetric exchange (Dzyaloshinskii-Moriya interaction vectors) [15, 16], $\vec{B}$ is the external magnetic field, the $\hat{K}_j$ represent uniaxial anisotropy axes and the $\vec{n}_i$ are the normalized classical spin vectors, $\vec{m}_i = \mu_i\vec{n}_i$. The neighbour relations for the pairwise interactions are visualised in terms of shells in Figure I.6.

The last term of Equation (I.1) describes the contributions from the point dipole approximation of the magnetic field of the spins, in micromagnetism often referred to as stray field interaction or demagnetising field interaction. Due to their long-range nature, the numerical cost of the double summation over all spins scales proportionally to N^2, which quickly becomes prohibitively expensive. More efficient methods of calculating dipolar interactions have been implemented in *Spirit* under the supervision of the author (see also Section 3.2).

The Hamiltonian (I.1), as it is implemented in *Spirit*, is not limited to any given number of neighbour shells for the exchange and Dzyaloshinskii-Moriya interactions. The list of interaction pairs can be specified arbitrarily, meaning that both long-range

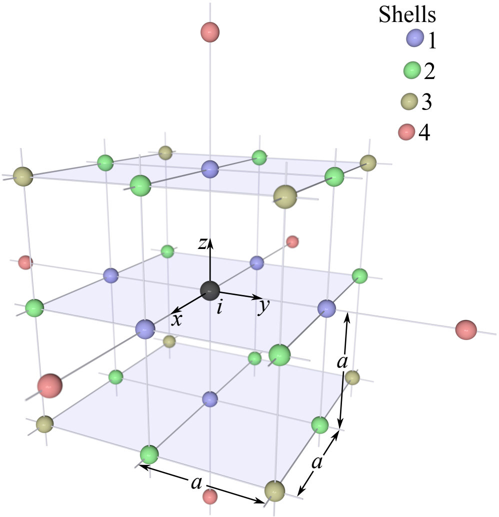


FIGURE I.6 – Visualisation of neighbour relations on a simple cubic lattice

Simple cubic lattice with lattice constant a. The colour of the spheres indicates which neighbouring atoms of the atom i are symmetry-equivalent. Only the first four shells are shown. Image provided by Nikolai Kiselev (see also [71]).

and non-isotropic interactions can be used. The geometry of a system can be freely chosen by defining a basis cell (containing any number of atoms), the Bravais vectors and the number of repetitions along them. Individual spins, as well as entire boundary layers can be pinned and defects (e.g. vacancies) and atoms of different types can be inserted explicitly or randomly.

In addition to Equation (I.1), a higher-order quadruplet interaction term [72], given by

$$E_{\mathrm{Quad}} = -\sum_{ijkl} K_{ijkl} \left(\vec{n}_i \cdot \vec{n}_j\right)\left(\vec{n}_k \cdot \vec{n}_l\right) \ , \tag{I.2}$$

is implemented in *Spirit*. The 4-spin-2-site [73] (or biquadratic, see Figure I.7a)), the 4-spin-3-site [74] (see Figure I.7b)), and the 4-spin-4-site [75] (also called "4-spin", see Figure I.7c)) interactions can all be represented as subsets of these quadruplets. Though they are implemented for completeness of the Hamiltonian model, they are not discussed in this work and second order derivatives were neither performed nor implemented. Further interaction terms can easily be added to the Heisenberg Hamiltonian in *Spirit*, as well as other models for the energy, such as the non-collinear Alexander-Anderson [76, 77] and micromagnetic models, as the Hamiltonian has a well-defined interface of functions.

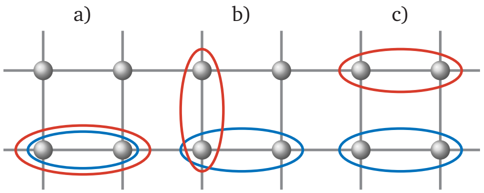


FIGURE I.7 – Quadruplet interactions

The quadruplet interactions (I.2) in *Spirit* are a general expression and can be used to represent a) 4-spin-2-site (biquadratic) b) 4-spin-3-site and c) 4-spin-4-site "4-spin" interactions. The ellipses visualise the scalar products between spins in these three cases.

I.2.2 *The micromagnetic Hamiltonian*

The next level of approximation beyond the extended Heisenberg model described in Section I.2.1 is given by the micromagnetic model [78–80], which allows the treatment of systems of a significantly larger scale. As this work repeatedly refers to the micromagnetic model and certain results are compared to it, this subsection will introduce some basic ideas and equations. The approach is to assume a continuum magnetisation vector field $\vec{m}(\vec{r})$, defined at any point $\vec{r}$ in space. It is not uncommon to perform analytical calculations, even for complex non-collinear structures such as domain walls [78, 81] or skyrmions [7, 32–34]. Note that an implementation of a micromagnetic Hamiltonian into *Spirit* was outside the scope of this work, but will likely be made in the future.

The continuum representation of the vectorfield allows using differential calculus on these systems, for which a very developed mathematical basis exists, which in many cases can provide analytical solutions. The possibility of finding exact solutions by applying calculus is arguably the main advantage of the micromagnetic approach over atomistic calculations. Amongst many other applications, It is useful for comparing to other results, obtained for example by numerical methods, or to generalise findings to a larger parameter space which would be costly to explore computationally.

Another advantage of treating magnetic systems in a continuum limit is the analytical description of the topology of the magnetic textures. In a 2D system, there is a well-defined topological winding number often used to characterise magnetic systems, called the topological charge [82]. For a unit vector field $\vec{n}(x, y)$ in the xy-plane, the charge can be written

$$Q = \frac{1}{4\pi} \int \left(\vec{n} \cdot (\partial_x \vec{n} \times \partial_y \vec{n}) \right) dx dy \,. \tag{I.3}$$

As skyrmions carry a topological charge, it can be used to calculate the number of skyrmions in a system.

The micromagnetic model assumes a continuous magnetisation distribution in space and time and the equivalent of the Hamiltonian is the total energy functional [83, 84]

$$\mathcal{F}[\vec{r}] = \int_V \mathcal{E}(\vec{r}) d\vec{r}\,, \tag{I.4}$$

with the energy density $\mathcal{E}$ with respect to the energy $\mathcal{E}_0$ of the collinear state. It can be written as an expansion in terms of magnetisation orientation $\vec{n}(\vec{r})$ and spatial derivatives $\partial\vec{n}(\vec{r})/\partial\vec{r}$ to the most common orders, corresponding to terms in the atomistic Hamiltonian (I.1), as

$$\mathcal{E}(\vec{r}) = \mathcal{E}_0 - M_\mathrm{S} B_\alpha^\mathrm{ext} n_\alpha + n_\alpha \mathcal{K}_{\alpha\beta} n_\beta + \mathcal{D}_{\alpha\beta\gamma}\left(n_\alpha \frac{\partial n_\beta}{\partial r_\gamma} - n_\beta \frac{\partial n_\alpha}{\partial r_\gamma}\right) + \frac{\partial n_\alpha}{\partial r_\beta} \mathcal{A}_{\beta\gamma} \frac{\partial n_\alpha}{\partial r_\gamma}\,, \tag{I.5}$$

where $\vec{n} = \vec{M}/M_\mathrm{s}$ is the orientation of the magnetisation at the point $\vec{r}$, M_s is the saturation magnetisation, $\vec{B}$ is an external magnetic field, $\mathcal{K}$ is the anisotropy tensor, $\mathcal{D}$ is a representation of the so-called spiralisation tensor (generally dependent on the magnetisation direction $\vec{n}(\vec{r})$) and $\mathcal{A}$ is the exchange stiffness tensor. Note that the exchange and spiralisation tensors can be written in various different forms and simplifications for common underlying lattice geometries exist [7, 85].

While in most cases, such as simple cubic or hexagonal distributions of points, a simple finite differences scheme suffices for the calculation of the spatial gradients $\partial\vec{n}/\partial\vec{r}$, for a more general implementation, a more complicated finite element method would be required. However, even the implementation of finite difference schemes is not necessarily trivial, as for example non-periodical boundary conditions require additional attention [86].

For the most common cases of isotropic magnets, where the tensors $\mathcal{K}$, $\mathcal{A}$ and $\mathcal{D}$ reduce to scalar constants, the relation to the atomistic model is given by [7–11]

$$M_\mathrm{s} = \mu_\mathrm{s}/a^3, \quad \mathcal{K} = K/a^3, \quad \mathcal{A} = J/(2a), \quad \mathcal{D} = D/a^2\,, \tag{I.6}$$

where a is the lattice constant and μ_s is the magnetic dipole moment of each spin.

These expressions can be used to obtain useful analytical solutions. For the isotropic case,

$$B_\mathrm{D} = \frac{\mathcal{D}^2}{2M_\mathrm{S}\mathcal{A}} = \frac{D^2}{\mu_\mathrm{S} J}, \quad L_\mathrm{D} = 4\pi\frac{\mathcal{A}}{\mathcal{D}} = 2\pi a \frac{J}{D}, \quad K_0 = \frac{a^3\mathcal{D}^2}{4\mathcal{A}} = \frac{D^2}{2J}\,, \tag{I.7}$$

where B_D is the critical field for the transition from the spin spiral to the collinear configuration as the ground state, l_D is the period of the incommensurate spin spiral and K_0 is the reference anisotropy. These can be used to increase the generality of the results of atomistic calculations by rescaling e.g. the domain size or the external magnetic field.

I.2.3 *Derivatives of the energy*

Due to the fact that magnetic systems are usually characterized by the orientation of the magnetic moments, while the length of the magnetic moments is assumed to be fixed and can be obtained from self-consistency calculations [77], the system is effectively constrained to a physical space $\mathcal{M}^{\text{phys}}$, given by a Riemannian manifold which is the direct product of spheres

$$\mathcal{M}^{\text{phys}} = \bigotimes_{i}^{N} \mathbb{S}^2 \subset \mathbb{R}^{3N} \,, \tag{I.8}$$

where $\mathbb{S}^2$ is a two-dimensional spherical surface. The notation will be that spin configurations $M = \{\vec{n}_i\} \in \mathcal{M}^{\text{phys}}$ are points on the manifold. This means that special care has to be taken when calculating derivatives [87, 88]. An extrinsic view onto this manifold [89] has proven highly convenient, avoiding the use of spherical coordinates and the undefined points and corresponding divergencies therein. Extrinsic here means that the physical manifold of possible spin configurations $\mathcal{M}$ is considered embedded into a surrounding space $\mathcal{E} = \mathbb{R}^{3N}$. As the Hamiltonian is defined in the entire embedding space, we can make use of derivatives calculated there, i.e. without any notion of constraints. In fact, it can be seen that methods, which use only first order derivatives, can simply use the unconstrained derivatives and – if needed – use a simple projection onto the tangent space to $\mathcal{M}^{\text{phys}}$. The constrained derivatives will be used to some extent in Chapter 1 and derived in completeness in Chapter 2, where the extrinsic view is covered in more detail.

While the default throughout this thesis is generally the embedding space representation (i.e. in $3N$ coordinates), to enhance clarity, or when this notation is more convenient, the space in which an object is represented is denoted by

$$\begin{aligned} &|^{3N} \text{ in the embedding space } \mathcal{E} \,, \\ &|^{2N} \text{ in the tangent space of } \mathcal{M} \,, \end{aligned} \tag{I.9}$$

where objects can be transformed as

$$\begin{aligned} \text{vectors } v|^{3N} &= T^{-1} v|^{2N} \,, \\ \text{matrices } A|^{3N} &= T^{-1} A|^{2N} T \,, \end{aligned} \tag{I.10}$$

where T is an orthonormal basis of the tangent space to the manifold $\mathcal{M}$. This means it is a matrix with the shape $3N \times 2N$ and is unitary, i.e. it has the property $T^{-1} = T^{\dagger}$ (if it is a real matrix it is orthogonal, i.e. $T^{-1} = T^{T}$).

We will now derive the gradient and the Hessian matrix, meaning the first and second derivatives of the Hamiltonian model (I.1) with respect to the spin orientation, which are used throughout this thesis. Analogous derivations can be made for the continuum model (I.5). For a more compact and general derivation, which is less accessible and further from the implementation in *Spirit*, see Appendix C. We will make use of the Einstein notation, in which the most commonly used part of the Hamiltonian (I.1) has the form

$$\mathcal{H} = -\sum_i B_\alpha n_{i\alpha} - \sum_i \frac{1}{K_i}(K_{i\alpha} n_{i\alpha})^2 - \sum_{\langle ij\rangle} J_{ij} n_{i\alpha} n_{j\alpha} - \sum_{\langle ij\rangle} D_{ij\alpha} \varepsilon_{\alpha\beta\gamma} n_{i\beta} n_{j\gamma}\,, \tag{I.11}$$

where $\vec{K}_i$ is the uniaxial anisotropy with $K_i = |\vec{K}_i|$ and we used the Levy-Civita symbol $\varepsilon_{\alpha\beta\gamma}$ to express the cross product of $\vec{n}_i$ and $\vec{n}_j$. We denote $\mathcal{H}$ the Hamiltonian defined on $\mathcal{M}^{\text{phys}}$ and its smooth continuation to the embedding space $\mathcal{E}$ as $\bar{\mathcal{H}}$.

The energy gradient in the embedding space has, in this notation, the form

$$(\nabla \bar{\mathcal{H}})_{i\alpha} = \frac{\partial}{\partial n_{i\alpha}} \bar{\mathcal{H}} = -B_\alpha - 2\frac{1}{K_i}(K_{i\alpha}^2 n_{i\alpha}) - J_{ij} n_{j\alpha} - \epsilon_{\alpha\beta\gamma} n_{j\beta} D_{ij\gamma}\,, \tag{I.12}$$

where we derived the cross product via

$$\frac{\partial}{\partial n_{i\alpha}} \left(D_{ij,\gamma} \epsilon_{\gamma\nu\beta} n_{i\nu} n_{j\beta} \right) = D_{ij,\gamma} \epsilon_{\gamma\nu\beta} \delta_{\alpha\nu} n_{j\beta} = \epsilon_{\alpha\beta\gamma} n_{j\beta} D_{ij,\gamma}\,. \tag{I.13}$$

The Hessian matrix in the embedding space $\mathcal{E} = R^{3N}$ is written

$$\bar{H}_{i\alpha j\beta} = \text{Hess}(\bar{\mathcal{H}})|^{3N} = \frac{\partial^2 \bar{\mathcal{H}}}{\partial n_{i\alpha} \partial n_{i\beta}}\,. \tag{I.14}$$

Since the Zeeman term of the Hamiltonian is linear in $\vec{n}$, the contribution of the external magnetic field to the Hessian is zero. The remaining terms we consider are quadratic in $\vec{n}$. The uniaxial anisotropy has a contribution

$$\bar{H}^{\text{ani}}_{i\alpha j\beta} = -\delta_{ij} \delta_{\alpha\beta}\ 2\frac{1}{K_i}(K_{i\alpha})^2\,. \tag{I.15}$$

The exchange interaction obviously gives

$$\bar{H}^{\text{exc}}_{i\alpha j\beta} = -\delta_{\alpha\beta}\ J_{ij}\,. \tag{I.16}$$

Finally, for the DM interaction, we use that

$$\frac{\partial}{\partial n_{j\beta}} \left(\epsilon_{\alpha\nu\mu} D_{ij,\nu} n_{j\mu} \right) = \epsilon_{\alpha\nu\mu} D_{ij,\nu} \delta_{\beta\mu} = \epsilon_{\alpha\nu\beta} D_{ij,\nu} \tag{I.17}$$

and write

$$\bar{H}^{\text{dmi}}_{i\alpha j\beta} = -\epsilon_{\alpha\nu\beta} D_{ij,\nu}\,, \tag{I.18}$$

which gives us only off-diagonal terms. The complete Hessian is thus written

$$\bar{H}_{i\alpha,j\beta} = -\delta_{ij}\delta_{\alpha\beta}\ 2\frac{1}{K_i}(K_{i\alpha})^2\ -\delta_{\alpha\beta}\ J_{ij}\ -D_{ij}\epsilon_{\alpha\nu\beta}D_{ij,\nu}\,, \tag{I.19}$$

which means that we get off-diagonal exchange and DMI contributions ($J_{ij} = D_{ij} = 0$ for $i = j$) and on-diagonal anisotropy contributions. We thus get the following two sub-matrices:

The spin-spin matrix blocks of the Hessian $\bar{H} = \partial^2\bar{\mathcal{H}}$ become:
diagonal blocks per anisotropy axis:

$$\bar{H}_{i=j} = -2\frac{1}{K_i}\begin{pmatrix} K_{ix}K_{ix} & K_{ix}K_{iy} & K_{ix}K_{iz} \\ K_{iy}K_{ix} & K_{iy}K_{iy} & K_{iy}K_{iz} \\ K_{iz}K_{ix} & K_{iz}K_{iy} & K_{iz}K_{iz} \end{pmatrix}, \tag{I.20}$$

off-diagonal blocks without dipolar interactions:

$$\bar{H}_{i\neq j} = \begin{pmatrix} -(J_{ij}+J_{ji}) & (-\vec{D}_{ij}+\vec{D}_{ji})_z & -(-\vec{D}_{ij}+\vec{D}_{ji})_y \\ -(-\vec{D}_{ij}+\vec{D}_{ji})_z & -J_{ij}-J_{ji} & (-\vec{D}_{ij}+\vec{D}_{ji})_x \\ (-\vec{D}_{ij}+\vec{D}_{ji})_y & -(-\vec{D}_{ij}+\vec{D}_{ji})_x & -J_{ij}-J_{ji} \end{pmatrix}. \tag{I.21}$$

The terms are obviously easy to calculate, but since the Hessian matrix is of size $3N_S \times 3N_S$, it becomes increasingly difficult to handle with growing system size. Calculating even just a few eigenvectors of the Hessian can thus become computationally very expensive. When using *Spirit*, the size of this matrix can also lead to problems with memory consumption, where the implementation of a sparse matrix alternative would be highly beneficial for cases without dipole-dipole interactions, as only $(N_{\text{neighbours}} + 1)/N_{\text{spins}}$ blocks can be non-zero. For a simple cubic lattice of 200×200 spins with only nearest-neighbour interactions, this would mean only 0.0125% of the entries could be non-zero. Finite-difference schemes for which the matrix need not be calculated in its entirety may also prove beneficial to the treatment of these matrices for larger systems, especially when including dipolar interactions.

Note that for the dipole-dipole interaction term, we simply get another constant contribution to the off-diagonal blocks

$$\bar{H}^{\text{ddi}}_{i\alpha j\beta} = \frac{\mu_0\mu_i\mu_j}{4\pi}\frac{\delta_{\alpha\beta} - 3r_{ij\alpha}r_{ij\beta}}{|\vec{r}_{ij}|^3}\,. \tag{I.22}$$

I.3 MONTE CARLO

To simulate thermodynamic properties of spin systems, such as the specific heat, magnetic susceptibility or critical temperature, one can use the well-established Monte Carlo (MC) method,[7] which is based on sampling a region of phase space [91]. For a very comprehensive description of Monte Carlo methods, see Ref. [92]. In *Spirit* the Metropolis Monte Carlo algorithm has been implemented, with an adaptive cone angle [55, 93] for the step size of the displacements.

In order to perform Metropolis Monte Carlo simulations, only the energy of the system needs to be calculated, making it the most straightforward method presented in this thesis (see also Table I.1). While the micromagnetic model describes the limit of $T = 0$, the well-known and precisely defined Monte Carlo algorithm, applied to an atomistic spin lattice model, offers an easy recipe for the calculation of temperature-dependent equilibrium properties.

The Hamiltonian (I.1) can readily be evaluated for single spins, meaning that the energy change for the displacement of a single spin is cheap to evaluate, unless interactions become long-ranged. Note that pair-wise interactions need to be counted twice in the calculation of a single spin displacement, as the change also affects the energies of the interacting spins.

Monte Carlo can for example be used to calculate the critical temperature T_c [94], which for a ferromagnetic system is the Curie temperature of the transition from the collinear phase at low temperature to the paramagnetic phase at high temperature. For antiferromagnets, T_c is the Néel temperature at which the antiferromagnetic order is lost. In general, as with other transitions, for example between non-collinear and paramagnetic states, the low-temperature order will at some point be destroyed by fluctuations, when the temperature is increased. Low-temperature ordering is described, by definition, by an order parameter, which needs to be defined to adequately characterise the problem at hand.

The critical temperature in a ferromagnet is related to the value of the exchange constant and the lattice structure, e.g. the dimension of the system and the number of neighbours. In the nearest neighbour mean field approximation [95] it can be expressed as

$$T_\mathrm{c} = \frac{zJ_{ij}}{3k_\mathrm{B}}, \tag{I.23}$$

where z is the order number, which is in our model the number of unique nearest neighbour exchange pairs J_{ij} for a single atom. However, this approximation is known to be an over-estimation, as the correlations between spins are not accurately taken into consideration. Other approaches, such as the classical spectral density

[7] the name Monte Carlo – a reference to the city in Monaco – was coined by Metropolis et al. [90] who were among the first to apply this method

method [96] can be used to better estimate the critical temperature, where the high temperature expansion method [97] gives the well-known result for a simple cubic lattice [98]:

$$T_{\mathrm{c}} = 1.44 \frac{J_{ij}}{k_{\mathrm{B}}} \,. \tag{I.24}$$

For a simple cubic lattice with $J_{ij} = 1$, this gives a mean-field value of $T_{\mathrm{c}} \approx 23.2$ K, while the correct value should be close to $T_{\mathrm{c}} \approx 16.71$ K.

The average total magnetisation in dependence on the temperature is best described by

$$M(T) = \begin{cases} (1 - T/T_{\mathrm{c}})^{b}, & T < T_{\mathrm{c}} \\ 0 & \text{else} \end{cases} , \tag{I.25}$$

where b is the critical exponent.

It can be derived [91] that the *linear response* of the susceptibility is given by

$$\chi = \frac{\partial \langle M^{\mathrm{tot}} \rangle}{\partial B^{\mathrm{ext}}} = \frac{1}{k_{\mathrm{B}} T} \left[\langle (M^{\mathrm{tot}})^2 \rangle - \langle M^{\mathrm{tot}} \rangle^2 \right] \tag{I.26}$$

and of the specific heat by

$$C_{\mathrm{V}} = \frac{\partial \langle E \rangle}{\partial T} = \frac{1}{k_{\mathrm{B}} T^2} \left[\langle E^2 \rangle - \langle E \rangle^2 \right] . \tag{I.27}$$

Note that the averages $\langle \ldots \rangle$ denote thermal averages, meaning that the observable values should be calculated many times at each temperature in order to give physical results. The total magnetization should be calculated as $M^{\mathrm{tot}} = \sqrt{\left(\sum_i \vec{m}_i\right)^2}$. Both χ and C_{V} should diverge at $T = T_{\mathrm{c}}$, making them suitable for correctness checks. The specific heat C_{V} is especially useful for identifying a phase transition, when the order parameter is not known.

These above quantities can be used for the numerical calculation of T_{c}. Fig. I.8 shows the results of the mentioned observables for a Monte Carlo calculation of a cube of size 30 × 30 × 30 with an exchange coupling of $J = 1$ meV.[8] The results demonstrate the validity of the implementation, as the expected critical temperature of $T_{\mathrm{c}} \approx 16.71$ K is matched with an error of less than 1%.

[8] These calculations were performed in part by Daniel Schürhoff, using Python scripts created by the author.

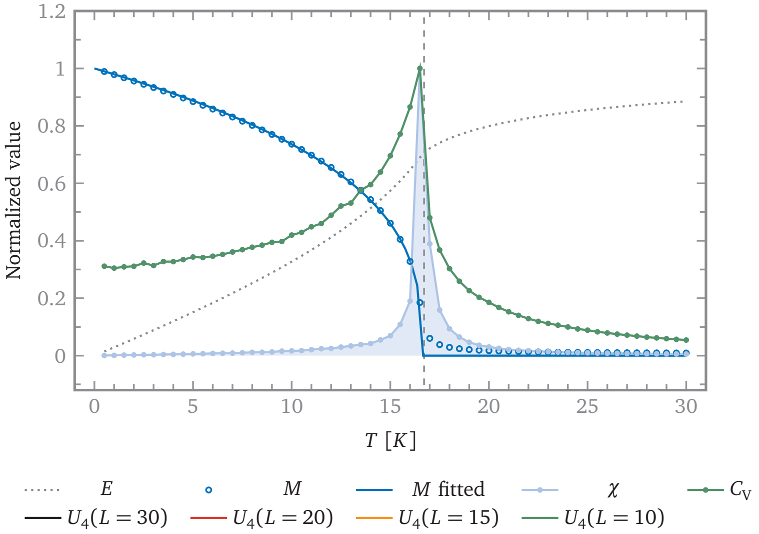


FIGURE I.8 – Monte Carlo test system

Normalized values of the total magnetization M, susceptibility χ, specific heat C_V and 4th order Binder cumulant U_4. The expected critical temperature is $T_c \approx 16.71$ K, while the value given by the observables is $T_c \approx 16.60$ K – an agreement within 1%. The exponent is fitted with $b = 0.33$. The system used is a 30 × 30 × 30 ferromagnet with $J = 1$ meV. At each temperature, 10k thermalisation steps were made before taking 100k samples. This graph has been published in [47].

A further thermodynamic observable is the fourth-order cumulant U_4, also known as Binder cumulant [99], for ferromagnets given by

$$U_4 = 1 - \frac{\langle (M^{\text{tot}})^4 \rangle}{3 \, \langle (M^{\text{tot}})^2 \rangle^2}, \tag{I.28}$$

where again, $\langle \ldots \rangle$ denotes thermal averaging. This quantity is an order parameter, which can be used to calculate the critical temperature of systems with second-order phase transitions, while accommodating for finite size scaling effects. U_4 should start at a value of $2/3$, when below T_c and change to a value of $4/9$ above T_c. By calculating the intersection of the curves of U_4 for different system sizes, a precise value for T_c can be obtained.

To make certain that T_c estimation in Figure I.8 is correct and to illustrate this method, the calculation is performed for additional cubes of size 25^3 and 20^3 atoms and the intersection point of the Binder cumulant curves is determined. This is shown in Figure I.9.

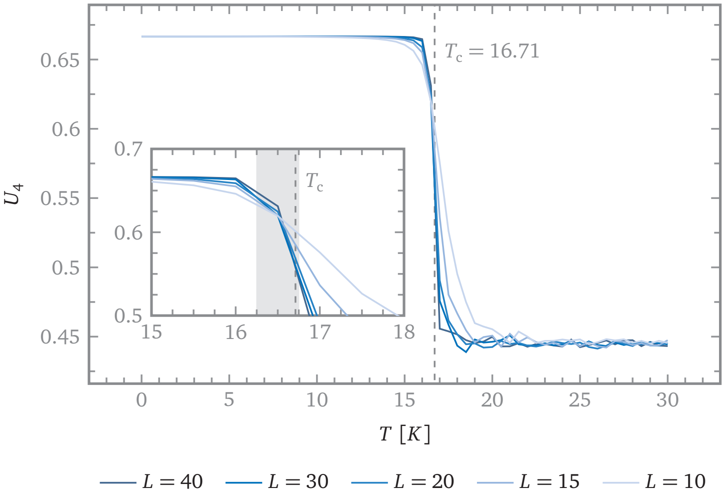


FIGURE I.9 – Monte Carlo test system

The 4th order Binder cumulant U_4 is shown for parameters as in Figure I.8 and different system sizes L. The inset shows a zoom on the region where the Binder cumulants intersect (16.25 K and 16.75 K, marked in grey). The expected critical temperature $T_c = 16.71$ therefore lies within the precision of these results, as the temperature step was chosen to be 0.5 K. This data in this graph has been published in [47].

This kind of calculation is a basic example of how the Python application programming interface (API) of *Spirit* can be used to easily extract the desired observables. This is an important part of this work, making atomistic calculations accessible and easy to run and will be detailed further in Chapter 3. The usefulness is exemplified by the following Python script, similar to what was used to obtain Figure I.8 and Figure I.9:

```
sample_temperatures = np.linspace(T_start, T_end, num=n_temperatures)

# Parameters
n_thermalisation = 10000
n_decorrelation = 10 # decorrelation between samples
n_samples = 100000

with state.State(input_file) as p_state: # State setup
    # Ferromagnet in z-direction
    configuration.plus_z(p_state)
    # Loop over temperatures
    for iT, T in enumerate(sample_temperatures):
        parameters.mc.set_temperature(p_state, T)

        # Cumulative average variables
        E = 0
```

```
            E2 = 0
            M = 0
            M2 = 0
            M4 = 0

            # Thermalisation
            parameters.mc.set_iterations(p_state, n_thermalisation, n_thermalisation) # We want n_thermalisation iterations and only a single log message
            simulation.start(p_state, MC) # Start a MC simulation

            # Start a single—shot MC simulation (we manually trigger each new iteration)
            simulation.start(p_state, MC, single_shot=True)
            # Sampling at given temperature
            for n in range(n_samples):
                # Run decorrelation
                for _ in range(n_decorrelation):
                    simulation.single_shot(p_state) # one MC iteration
                # Get energy
                E_local = system.get_energy(p_state) / NOS
                # Get magnetization
                M_local = np.array(quantities.get_magnetization(p_state))
                M_local_tot = np.linalg.norm(M_local)
                # Add to cumulative averages
                E += E_local
                E2 += E_local2
                M += M_local_tot
                M2 += M_local_tot2
                M4 += M_local_tot4
            # Make sure the MC simulation is not running anymore
            simulation.stop(p_state)

            # Average over samples
            E /= n_samples
            E2 /= n_samples
            M /= n_samples
            M2 /= n_samples
            M4 /= n_samples

            # Calculate observables
            chi = (M2 — np.dot(M, M)) / (constants.k_B  T)
            c_v = (E2 — E2) / (constants.k_B  T2)
            cumulant = 1 — M4 / (3  M22)
```

LISTING 1 –

Using the Monte Carlo method to sample the magnetisation and other observables. The length of the script is owed entirely to the usage of decorrelation steps and the number of different observables, which are sampled – the structure of the script is very simple.

Additional improvements on thermal sampling can be made, for example by using the Ziggurat algorithm [100] in thermal equilibration. Also, the parallel tempering algorithm has proven to be effective [101–103] and while it has not been implemented in *Spirit*, the usage of Python and a message passing interface (MPI) package would enable one to quite easily reproduce this algorithm in a Python script, using the API of *Spirit*.

There are a lot of other order parameters and observables, which can be calculated [63], but two are worth noting here. The first is the topological charge of a discretised 2D system (see also Equation (I.3) for the continuum equivalent)

$$Q = \frac{1}{4\pi} \sum_{\langle ijk \rangle} \Omega(\vec{n}_i, \vec{n}_j, \vec{n}_k)\,, \tag{I.29}$$

where $\langle ijk \rangle$ are unique triplets and Ω is the solid angle between a triplet of spins. It is often used as a measure for the number of skyrmions in a system, as they carry a topological charge – for the simple case of a skyrmion pointing downward in an upward homogeneous background the charge is $Q = -1$ (see also Figure 1.10 in Section 1.2). The second is the entropy

$$S = k_\mathrm{B} \log \Gamma\,, \tag{I.30}$$

where Γ is the phase space volume compatible with a certain thermodynamical state, and the free energy

$$F = \langle E \rangle - TS\,. \tag{I.31}$$

The entropy will appear again in transition rate calculations (see Section 1.3.3). As the specific heat can be calculated as $C_T = T\partial S/\partial T$ and we already know Equation (I.27), the entropy can be found by integration

$$S(T) = S(T_0) + \int_{T_0}^{T} \frac{C_T(T')}{T'} dT'\,. \tag{I.32}$$

For this, the entropy must be known at some initial temperature, where in systems satisfying the third law of thermodynamics, we can choose $T_0 = 0$, as $S(0) = 0$. However, if there is a continuous symmetry in the ground state, we cannot choose $T_0 = 0$, as $S(0) = -\infty$ in this case. In practice, one can then choose a large value $T \rightarrow \infty$ and an approximation for $S(\infty)$ and integrate downwards.

Spin-spin and other correlation functions can also give additional information, such as the magnon dispersion relation, but they are more complex to calculate, as they generally involve double sums. Since the implementation of fast Fourier transforms (FFTs) in *Spirit* for the calculation in the context of dipolar interactions (see Section 3.2), the sampling of correlation functions has also become feasible, as the double sums can be replaced by convolutions [92]. However, this has not yet been implemented.

I.4 LANDAU LIFSHITZ GILBERT DYNAMICS

In contrast to the Monte Carlo method, which has no time resolution, simulations based on the Landau-Lifshitz-Gilbert (LLG) equation allow the calculation of physical transfromations from one state into another over time. Moreover, MC calculations can be reproduced by the usage of thermal noise in the LLG equation. However, numerically solving the LLG equation is a more involved task, as it requires derivatives of the energy and the spin distribution (see also Table I.1).

The time evolution of the magnetisation can be calculated within DFT [66, 104] and time-dependent DFT (TD-DFT) methods have been developed in order to go beyond the adiabatic approximation of atomistic spin dynamics. However, the (stochastic) LLG is often employed for systems on the order of 10^4 to 10^9 magnetic atoms – a size for which the evaluation of the equations of motion of TD-DFT over several hundred ps is currently unfeasible. In contrast, atomistic spin dynamics of 10^4 spins using the LLG equation can even be performed on current day mobile phones.[9]

The implicit form of the Landau-Lifshitz-Gilbert [78, 105] equation including spin torque and temperature contributions can be written [106, 107]

$$\frac{\partial \vec{n}_i}{\partial t} = -\frac{\gamma}{(1+\alpha^2)\mu_i}\vec{n}_i \times \vec{B}_i^{\text{eff}} - \frac{\alpha-\beta}{(1+\alpha^2)}u\vec{n}_i \times (\hat{j}_e \cdot \nabla)\vec{n}_i + \vec{n}_i \times \frac{\partial \vec{n}_i}{\partial t}\,, \tag{I.33}$$

where the explicit form of the LLG equation (I.33) then reads

$$\begin{aligned}\frac{\partial \vec{n}_i}{\partial t} =& -\frac{\gamma}{(1+\alpha^2)\mu_i}\vec{n}_i \times \vec{B}_i^{\text{eff}} - \frac{\gamma\alpha}{(1+\alpha^2)\mu_i}\vec{n}_i \times (\vec{n}_i \times \vec{B}_i^{\text{eff}}) \\ &- \frac{\alpha-\beta}{(1+\alpha^2)}u\vec{n}_i \times (\hat{j}_e \cdot \nabla)\vec{n}_i + \frac{1+\beta\alpha}{(1+\alpha^2)}u\vec{n}_i \times (\vec{n}_i \times (\hat{j}_e \cdot \nabla)\vec{n}_i)\,,\end{aligned} \tag{I.34}$$

where $\mu_\text{B} \approx 0.05788\frac{\text{meV}}{\text{T}}$ is the Bohr magneton and $\gamma = g_e\mu_\text{B}/\hbar \approx 0.1761\frac{\text{rad}}{\text{ps T}}$ is the electron gyromagnetic ratio, α is the scalar Gilbert damping parameter, $\vec{m}_i = \mu_i\vec{n}_i$, μ_i is the spin magnetic moment, β is a non-adiabatic parameter and $\vec{j}$ is the current density vector. The spin current strength is given by $u = Pg_e\mu_\text{B}/2eM_\text{s}$, where P is the polarisation, g_e the Landé g-factor, M_s the saturation magnetisation and e the electron charge magnitude. $\vec{B}^{\text{eff}} = -\nabla\mathcal{H}$ is the effective magnetic field.

The terms in the explicit LLG equation (I.34) correspond to the precession and damping of the spin in an effective magnetic field – which may include stochastic forces – as visualised in Figure I.10 and the precession-like and damping-like spin current induced torque terms.

Note that the gradient approximation can be transferred to a thin layer approximation with orthogonal current [109–113] as $(\hat{j}_e \cdot \nabla)\vec{n}_i \rightarrow \vec{P}$ with $\vec{P}$ the polarisation

[9] See the web UI of *Spirit* at `juspin.de`

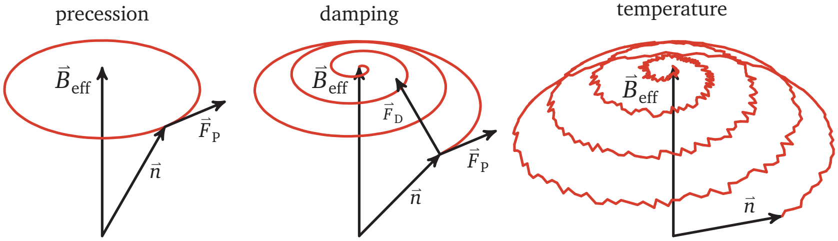


FIGURE I.10 – Schematic of spin movement under LLG equation

Side by side: the movement of a single spin in an effective field under the LLG equation. Left: only precession. Middle: also damping. Right: also stochastic noise. Images adapted from [108].

direction. See Ref. [114] for more details on spin current torques and the calculation of $(\hat{j}_e \cdot \nabla)\vec{n}_i$.

In this notation for the LLG equations (I.33) and (I.34), the effective field $\vec{H}_i$ may contain also e.g. a stochastic thermal field (see also Figure I.10) and is therefore written

$$\vec{H}_i^{\text{eff}} = -\frac{\partial \mathcal{H}}{\partial \vec{n}_i} + \vec{H}_i^{\text{ther}}, \quad \vec{H}_i^{\text{ther}} = \sqrt{2\alpha \frac{\mu_i}{\gamma} k_{\text{B}} T \vec{\xi}_i}\,, \tag{I.35}$$

where the vectors $\vec{\xi}_i$ consist of three standard normally distributed and independent random values each and the amplitude of the fluctuating thermal field is according to the fluctuation-dissipation theorem [115–117]. Note that in time-integration schemes, to fulfill the fluctuation-dissipation relation, the thermal field needs to be normalized by the time step with a factor $1/\sqrt{\delta t}$. For more details on the integration of the stochastic LLG equation see for example references [118, 119] and references therein.

Note that the LLG equation restricts the movement of the spins to the surface of a sphere, as all terms, due to the vector products with $\vec{n}_i$, are orthogonal to the spin $\vec{n}_i$ (see Figure I.10). This becomes plainly obvious when rewriting the LLG equation as

$$\frac{\partial \vec{n}_i}{\partial t} = \vec{n}_i(t) \times \vec{A}(t, \{\vec{n}_i(t)\})\,. \tag{I.36}$$

This implies that only the component of the effective field, which is orthogonal to the spin, acts on it:

$$\frac{\partial \mathcal{H}}{\partial \vec{n}_i} \to \frac{\partial \mathcal{H}}{\partial \vec{n}_i} - \left(\frac{\partial \mathcal{H}}{\partial \vec{n}_i} \cdot \vec{n}_i\right)\vec{n}_i\,. \tag{I.37}$$

This corresponds to the physical restriction of the spin orientation onto the surface of a unit sphere, mentioned in Section I.2. When implementing the numerical methods to solve the system of coupled ordinary differential equations (ODEs) (I.36), this constraint on the spin length must be considered. The details of the implementation in *Spirit* are given in Section 3.3. Currently, Heun's method [55], a 4th order Runge-

Kutta solver, Mentink's semi-implicit method B (SIB) [118] and Depondt's Heun-like method [120] are implemented.

I.4.1 *Precession dynamics*

A simple example of the LLG equation is the precession-only case (see Figure I.10). The analytical solution for a single spin in a magnetic field is readily calculated and for the initial condition $n_x = 1$, is determined by the angle of the spin with respect to the x-axis

$$\varphi(t) = \frac{\gamma}{(1+\alpha^2)\mu}|\vec{B}|t\,. \tag{I.38}$$

The calculation for this example is shown in Figure I.11.

Evidently and as expected, the spin rotates in the xy-plane, without changing its z-component within the numerical precision over a time span of 500 ps. This analytical equation can be compared to the numerical result produced by the various solvers implemented in *Spirit* (see also Section 3.3) in order to determine a measure of precision of the numerical solution. The results for the Depondt solver, shown in Figure I.11, indicate that the spin is kept perfectly in the xy-plane. However, the precession frequency is not met exactly and the error of the spin $\vec{n}$ therefore increases linearly over time, as it behaves additively.

We now take damping into account, so that the z-component becomes time-dependent and the solution reads

$$\begin{aligned}
n_z(t) &= \tanh\left(\frac{\alpha\gamma}{(1+\alpha^2)\mu}|\vec{B}|t\right),\\
\varphi(t) &= \frac{\gamma}{(1+\alpha^2)\mu}|\vec{B}|t\,,\\
n_x(t) &= \cos(\varphi(t))\sqrt{1-n_z^2(t)}\,,\\
n_y(t) &= \sin(\varphi(t))\sqrt{1-n_z^2(t)}\,.
\end{aligned} \tag{I.39}$$

The simulation presented in Figure I.11 is repeated with a finite damping of $\alpha = 0.1$ and the results shown in Figure I.12.

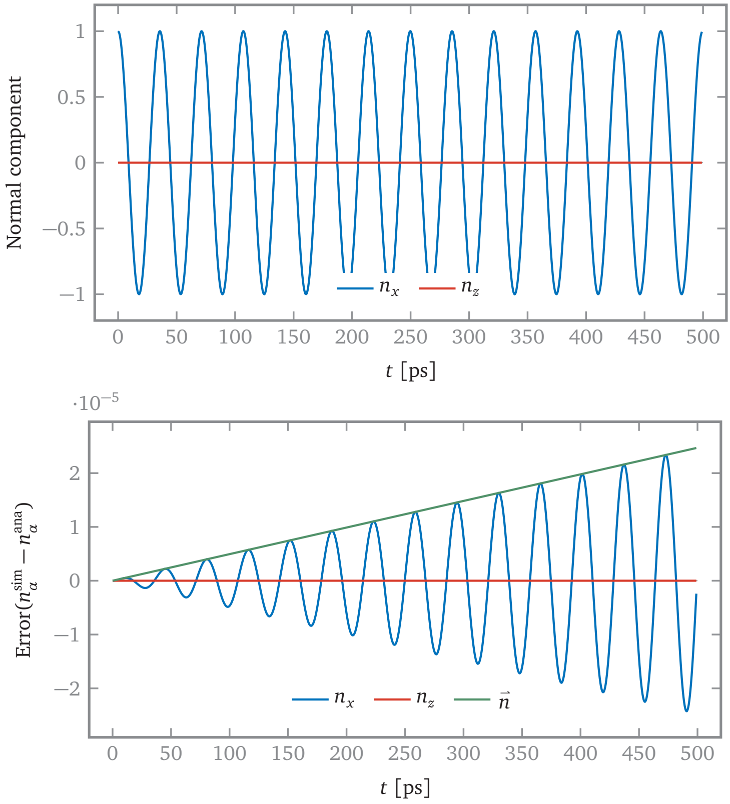


FIGURE I.11 – LLG dynamics: spin precession without damping

A single spin in a magnetic field of $B = 1$ T along z-direction, with zero damping $\alpha = 0$ and time step $dt = 0.01$ ps, simulated for a total time of 500 ps. The simulation is performed with the Depondt solver. Top: the components n_x and n_z of the undamped spin over time t. As there is no damping, the n_z remains constant, while the spin rotates in the xy-plane. Bottom: the error of the numerical simulation with respect to the analytically expected result. As there is a numerical imprecision at every time step, the total error is additive and rises linearly. The imprecision does not, however, rotate the spin out of the xy-plane.

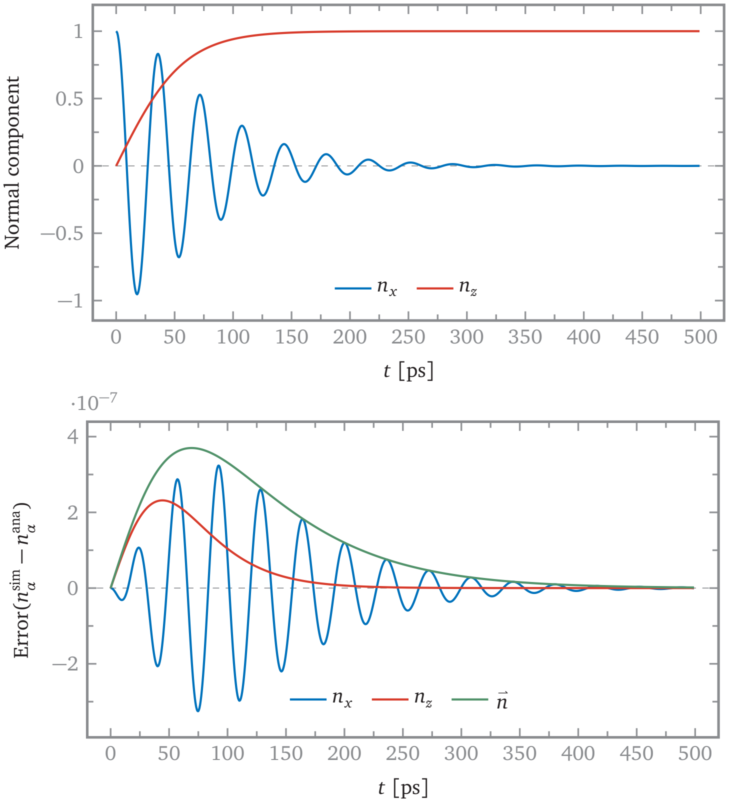


FIGURE I.12 – LLG dynamics: damped spin precession

A single spin in a magnetic field of $B = 1$ T, with a damping of $\alpha = 0.1$ and time step $dt = 0.01$ ps, simulated over a total time of 500 ps. The simulation is performed with the Depondt solver. Top: the precession and damping of the spin components n_x and n_z. The spin relaxes to the orientation of the external magnetic field. Bottom: the error of the numerical simulation with respect to the analytically expected result. While the error is additive and initially grows, it is limited by the fact that the spin converges against the analytical result and therefore converges down to zero for long times. This figure is adapted from [47].

The profiles appear as expected, showing a quickly damped spin oscillation and convergence to the orientation of the magnetic field. As the spin converges to the z-axis, the error, after an initial rise, converges to zero.

See Section 3.3 for additional tests and comparisons of the implemented solvers.

Note that using LLG dynamics simulations, various correlation functions can be calculated by time-dependent sampling (see Ref. [63] for an exhaustive overview over LLG-related methods), for example to extract spin wave excitation spectra [121]. A time-dependent field can be applied to the system and the response measured in order to find resonance peaks and thereby find eigenmodes of a system without more complicated approaches, such as those discussed in Section 1.5 – another benefit being the memory-efficiency of this kind of calculation compared to the huge matrices needed for the dynamical equation.

I.4.2 *Stochastic LLG*

Much like the Monte Carlo method, the stochastic LLG equation can be sampled in order to obtain results about a given system, such as the critical temperature T_c. The numerical integration of this stochastic equation requires some attention to details of the solver, as has been laid out in the literature [118, 119, 122, 123].

The calculation of the temperature induced transition of a small ferromagnetic cube to the paramagnetic phase, presented in Figure I.8, is reproduced here, though with significantly more samples, in Figure I.13.[10] The results agree well and give the same expected value with a deviation of 1%.

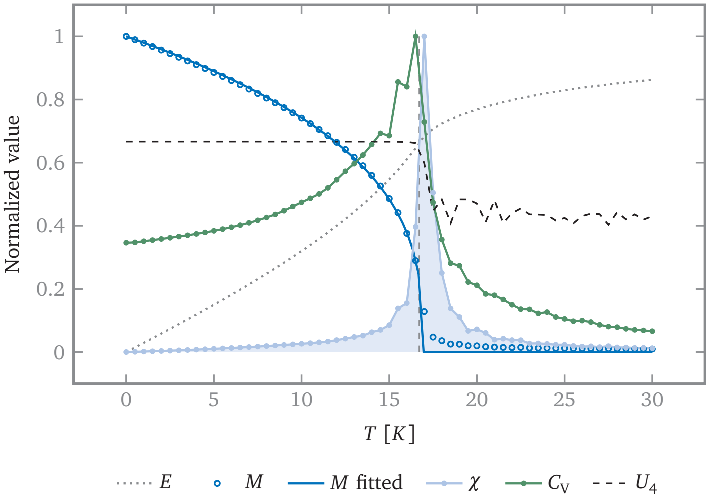


FIGURE I.13 – LLG temperature test system

The energy per spin E and normalized values of the total magnetization M, susceptibility χ, specific heat c_V and 4th order Binder cumulant U_4. The expected critical temperature is $T_c \approx 16.71$ K, while the value given by the observables is $T_c \approx 16.92$ K – an agreement of 1.2%. The exponent is fitted with $b = 0.33$. The system used is a $30 \times 30 \times 30$ ferromagnet with $J = 1$ meV. At each temperature, 100k thermalisation steps were made before taking 200k samples. This figure has been published in [47].

In general, Langevin dynamics can be used to obtain far more results, as it can –

[10] These calculations were performed in part by Daniel Schürhoff, using Python scripts created by the author.

in principle – reproduce all the observables which can be gathered by Monte Carlo (see Section I.3 and Ref. [63]). Though thermal sampling of static properties can be significantly more costly when using LLG instead of MC, the benefit is that one can sample time-dependent processes, such as the temperature gradient induced [124] propagation of domain walls [125, 126] or skyrmions [127].

I.4.3 *Current-driven dynamics*

A very important part of the LLG equation (I.34) are the current-induced precession-like and damping-like torques, which model the effect of an electric current onto the magnetisation structure [107]. With the ability to simulate these effects, the current-induced motion of magnetic configurations such as domain walls, and therefore potential racetrack memory designs, can be studied numerically [32, 128].

As an illustrative example and a proof of concept, the movement velocity of a domain wall in a head-to-head spin chain is shown in dependence on the applied current. The results from Ref. [107] for this dependence at different values of the non-adiabatic parameter β are reproduced in Figure I.14.[11] The chain is oriented along the x-axis and the first and the last spin are pinned to point in $+x$ and $-x$ direction respectively. As a special case of the more general Hamiltonian (I.1) of Section I.2, the Hamiltonian for this example can be written as follows:

$$\mathcal{H} = -\sum_i K_1 S_{ix}^2 + K_2 S_{iy}^2 - J \sum_{\langle ij \rangle} \vec{S}_i \cdot \vec{S}_j \,. \tag{I.40}$$

where $K_1 = 0.01$ meV and $K_2 = 0.005$ meV are the values for the anisotropy in x- and y-direction respectively. For details, see the reference [107].

The approximate prediction for the velocity, given in Ref. [32] $\langle v \rangle = \sqrt{u^2 - u_c^2}/(1+\alpha^2)$ fits the results shown in Figure I.14 well. As expected, we observe the Walker breakdown [109, 129] and a critical effective velocity of $u_c \approx 0.0414$, which is in close agreement with the reported value of $u_c \approx 0.0416$. Note, for $\beta = 0.1$ and currents larger than u_{W} and for $\beta = 0$ and currents larger than u_c, the domain wall starts rotating around the x-axis.

This method can of course also be applied to simulate the skyrmion racetrack concept. It is therefore directly related to calculations presented in the following Chapter 1 and Refs. [38, 130], where the thermal stability of a skyrmion inside a racetrack is estimated.

[11] The data was gathered by Constantin Disselkamp

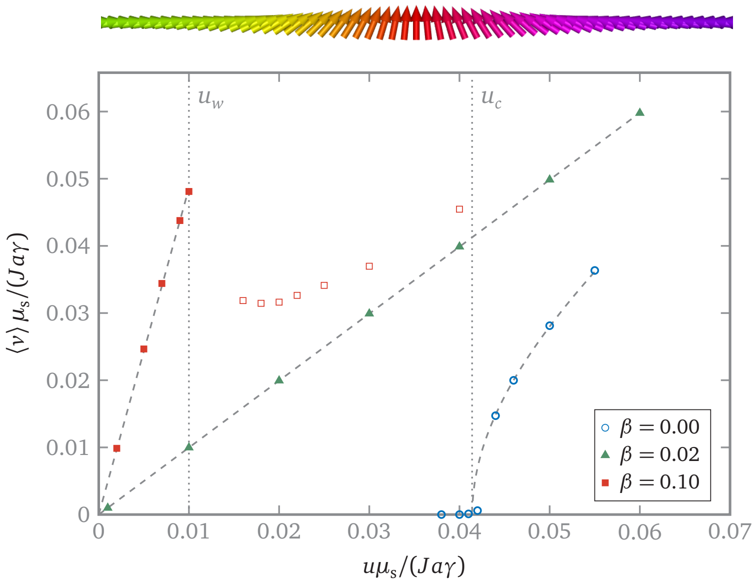


FIGURE I.14 – Domain wall motion under current

The average velocity of head-to-head domain wall (see top) for various values of the non-adiabatic parameter β. Damping is set to $\alpha = 0.02$. For $\beta = 0.10$ the Walker breakdown occurs at approximately $u_W \approx 0.01$. For $\beta = 0$ a critical current is at $u_c \approx 0.0414$. From this point the relation $\langle v \rangle = \sqrt{u^2 - u_c^2} / (1 + \alpha^2)$ mentioned by Thiaville *et al.* [32] takes effect. The mentioned relation is fitted to the data for $\beta = 0$. The dashed lines show linear fits and open symbols denote rotation around the x-axis. The results from ref. [107] are reproduced well. The data was gathered by Constantin Disselkamp and the figure is adapted from [114]. This figure has been published in [47].

To illustrate the interesting dynamics, which can be induced by spin polarised electric currents, the nucleation of a skyrmion from a notch in a racetrack [21] is qualitatively reproduced in Figure I.15.

It can be seen that the simulation of a racetrack configuration including a notch can easily be achieved and conveniently visualised in *Spirit* (see Section 3.4 on how this can be conveniently created from the GUI).[12] The first implementation of spin currents (in the monolayer approximation) into *Spirit* was performed in collaboration with the author by Daniel Schürhoff as a part of his Master thesis [131], who also performed some quantitative skyrmion racetrack simulations. The implementation of in-plane currents and further skyrmion racetrack simulations were performed by Constantin Disselkamp as part of his Bachelor thesis [114] under the supervision of the author.

[12] It is even easily possible to simulate spin currents on current mobile phones in the web user interface of *Spirit*

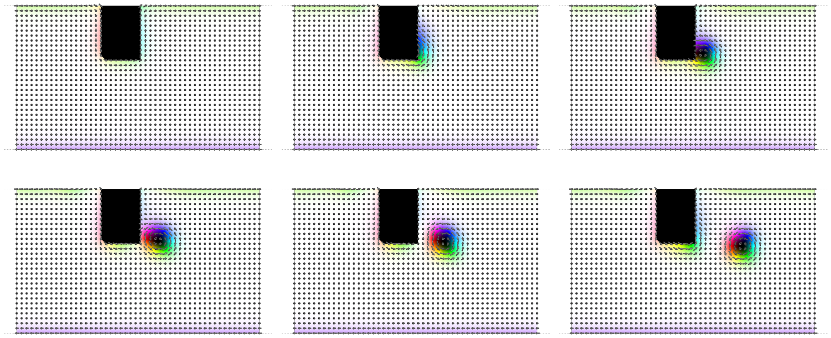


FIGURE I.15 – Skyrmion nucleation under current

The shown screenshots from the GUI of *Spirit* show the nucleation of a skyrmion from a notch (black area) in a monolayer racetrack configuration, analogous to what is presented in [21], with a current flowing along the track with periodical boundary conditions in the same direction. The system size was chosen to be 50×30 and the notch of size 7×11. Note the tilt along the boundaries of the track and the notch, which help the nucleation of the skyrmion. This illustrates the possibility to easily simulate such processes in *Spirit*.

I.4.4 *Energy minimisation*

In order to converge the system to a minimum of the energy landscape, various well known numerical methods can be used, such as the nonlinear conjugate gradients (NCG) method (see Section 3.3). However, a more straightforward approach can be taken by iteratively moving the system towards the direction of the effective field $\vec{B}_i^{\text{eff}} = -\partial\mathcal{H}/\partial\vec{n}_i$ using pseudodynamics. As long as the path towards the local minimum is not of interest, other dynamical equations can be used instead of the LLG equation. The methods for solving the coupled ordinary differential equation (ODE) can then also be used for energy minimisation. One option is to consider the damping term of the LLG equation (no precession term, stochastic fluctuations or spin polarised currents), which gives

$$\frac{\partial\vec{n}}{\partial t} \propto \vec{n} \times \left(\vec{n} \times \frac{\partial\mathcal{H}}{\partial\vec{n}_i}\right). \tag{I.41}$$

Since $|\vec{n}| = 1$, for any vector $\vec{v}$ it holds that $\vec{n} \times (\vec{n} \times \vec{v}) = \vec{v} - \vec{n}(\vec{n} \cdot \vec{v})$, i.e. only the component orthogonal to the spin is relevant.

Analogously, a Verlet-like velocity projection method can be used. The corresponding equation of motion gives

$$\frac{\partial^2\vec{n}}{\partial t^2} \propto \vec{n} \times \left(\vec{n} \times \frac{\partial\mathcal{H}}{\partial\vec{n}_i}\right). \tag{I.42}$$

This method has been implemented in *Spirit* (see Section 3.3) and has been seen to significantly improve the speed of convergence. Note that this velocity projection method needs to separately consider the normalisation of the spin vectors, as, even infinitessimally, it does not confine the movement of the spin to the surface of a sphere.

In all these cases, the system will converge to a minimum of the energy landscape. However, the result can be highly dependent on parameters, initial spin configuration and the method used. The direct minimisation with respect to a given force vector also finds application in other problems, such as the calculation of transition paths (see Section 1.2) or the search for saddle points of the energy landscape (see Chapter 2).

RATE THEORY AND MINIMUM ENERGY PATHS 1

"The career of a young theoretical physicist consists of treating
the harmonic oscillator in ever-increasing levels of abstraction."

— Sidney Coleman

To make the flow of this thesis complete and consistent, this chapter will cover rate theory methods, namely the geodesic nudged elastic band (GNEB) method to calculate transition paths and the harmonic transition state theory (HTST) to calculate transition rates. This will provide not only the background and motivation for the saddle point search method, developed as a part of this work and presented in Chapter 2, but also a more coherent description in the framework and notation of this thesis. We will see how these higher order methods build onto the foundation of the LLG dynamics and why there was a need to develop a method to search for new transition paths.

1.1 INTRODUCTION

The development of novel magnetic devices, for example in spintronic applications [41], can be greatly accelerated by the predictive potential of theoretical calculations. As laid out in the Introduction, there is a need to determine the lifetimes of magnetic states. The challenge in the estimation of lifetimes is to identify the various, physically possible transformations that a magnetic configuration can undergo and to estimate the probabilities for their occurrence at finite temperature. A problem faced by the Monte Carlo and stochastic LLG methods is the potential disparity between the simulated and the laboratory timescale, which can easily be minutes, hours and even significantly longer. Recall for example the challenge of designing materials in which magnetic skyrmions are small enough while being sufficiently stable at ambient temperature, meaning skyrmion lifetimes on the order of magnitude of at least days. Novel rate theory methods, different from the stochastic approaches shown in the Introduction, therefore become necessary – especially when calculations, such as presented in Ref. [54], become too time-consuming due to the rare occurrence of transition events. Spin ice systems are a good example for systems with rare transition events [132–134].

Transition state theory (TST) calculates statistical estimates of transition rates from the basic properties of the energy landscape without the need for any sampling (see Section 1.3 for details), but requires knowledge of the minimum energy paths (MEPs) (more precisely: the saddle points along the MEPs) corresponding to the possible transitions. A simple example would be the reversal of a magnetic domain, which can occur by homogeneous rotation or nucleation and subsequent travel of a domain wall. The methods presented in this chapter have very general applicability and are not limited to any assumptions about the orientations of the spins, such as collinearity.

In such cases, where final states are known, the geodesic nudged elastic band (GNEB) method [56, 77] can be used to find the MEPs of the transitions and, thereby, the activation energies. An illustration of such an MEP is shown in Figure 1.1. Note that the author has previously described the geodesic nudged elastic band (GNEB) in his master thesis [108].

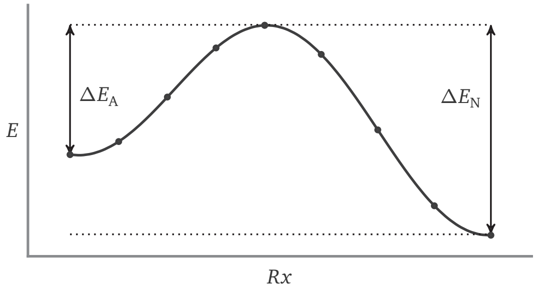


FIGURE 1.1 – Schematic energy barrier

An energy path $E(Rx)$ is shown schematically, with the annihilation barrier ΔE_A and the nucleation barrier ΔE_N of a metastable state with respect to the ground state. The path is discretised by a set of configurations, often referred to as images. Rx is the reaction coordinate, which is simply a measure of distance along the transition. Assuming a converged MEP $E(Rx)$, ΔE_A is the lowest energy barrier between the two states for this transition. Note that the energy landscape of a spin system may have high dimensionality, meaning the maximum of $E(Rx)$ is in general a saddle point of the energy landscape. Image adapted from [108].

The reaction coordinate Rx is defined as the distance along a path through configuration space. For a discrete set of points along the path, referred to as images, it can be simply defined as the cumulative sum of the distances between images:

$$Rx^{\nu} = \begin{cases} 0\,, & \nu = 1 \\ \sum\limits_{\mu=2}^{\nu} L^{\mu-1,\mu}\,, & \nu > 1 \end{cases}\,, \tag{1.1}$$

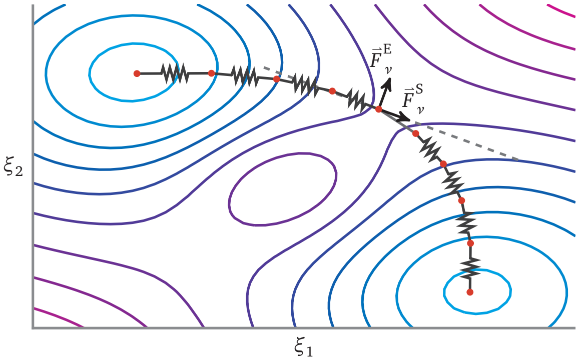


FIGURE 1.2 – Schematic chain of images in energy contour with GNEB forces

A schematic energy contour is shown as a contour plot in two coordinates ξ_1 and ξ_2 (imagine e.g. two angles of a spin). A path through the energy landscape is drawn with visualised springs between images, where the spring force $\vec{F}_{\mathrm{S}}$ and energy gradient force $\vec{F}_{\mathrm{E}}$ are visualised for one image. The spring force acts tangent to the path, to keep the distance between images equal, whereas the gradient force pulls the images towards lowest energy, orthogonal to the path. It is comparable to an elastic band with movable weights attached, that is laid onto a contoured surface. Image adapted from [108].

where μ and ν are indices of the discrete images along the chain and $L^{\nu\mu}$ is a measure of distance in configuration space between two images M^ν and M^μ.

1.2 CALCULATING MINIMUM ENERGY PATHS

1.2.1 *The nudged elastic band method*

The so-called nudged elastic band (NEB) method [135] was developed in the context of chemical reactions and molecular dynamics for the exact goals outlined in the previous section. It was subsequently generalised to be applicable to spin systems [56], where it is called geodesic nudged elastic band (GNEB) method. It has since been used for various 2D and 3D spin systems, for example to prove the metastability of a state with respect to a certain transition [38, 46, 47, 58, 61, 136].

The NEB method uses a discrete set of points along the path, called images, which are iteratively optimised a) towards lower energies orthogonal to the transition path and b) for equal distances along the discretised transition path, corresponding to an elastic band. To achieve this, specially constructed gradient and spring forces are applied to each configuration, as schematically visualised in Figure 1.2.

The total NEB force is written

$$F_\nu^{\mathrm{tot}} = F_\nu^{\mathrm{S}} + F_\nu^{\mathrm{E}}, \tag{1.2}$$

where ν is the image index along the chain, F^{S} is a spring force and F_ν^{E} is an energy gradient force. The forces in this section are $3N$-dimensional vectors.

A simple definition of the spring force, which will lead to convergence towards an equidistant distribution of images (in phase space) along the path, is given by

$$F_\nu^{\mathrm{S}} = (L_{\nu-1,\nu} - L_{\nu,\nu+1})\,\tau_\nu, \tag{1.3}$$

where $L_{\nu,\mu}$ is a measure of distance between images ν and μ, as in Equation (1.7), and τ_ν is the (normalised) path tangent at image ν.

As an alternative to the approach of spring forces exists, the images can instead be redistributed, for example linearly, along the geodesic path between the images after a given number of iterations. However, the author's experience has shown that this approach can exhibit worse behaviour than the spring forces in certain cases where the density of images is low, as when large steps in the configuration space may produce kinks in the path.

As previously stated, the gradient force F_ν^{E} should minimise the energy of the image in the degrees of freedom orthogonal to the path. This is easily achieved by removing the component of the gradient, which points along the path, i.e. by orthogonalising it with respect to the tangents:

$$F_\nu^{\mathrm{E}} = -\nabla E_\nu + (\nabla E_\nu \cdot \tau_\nu)\tau_\nu. \tag{1.4}$$

Note that the scalar product $\nabla E_\nu \cdot \tau_\nu$ is that of the $3N$-dimensional vectors.

The path tangents can be easily approximated by finite differences between the images, but in order to avoid the formation of kinks in the path, the definitions given in ref. [137] should be used. On slopes,

$$\tau_\nu = \begin{cases} \tau_\nu^+, & \text{if } E_{\nu+1} > E_\nu > E_{\nu-1} \\ \tau_\nu^-, & \text{if } E_{\nu+1} < E_\nu < E_{\nu-1} \end{cases}, \tag{1.5}$$

and, to ensure smooth transitions between rising and falling slopes, at minima and maxima of the path, i.e. if E_ν is an extremum

$$\tau_\nu = \begin{cases} \tau_\nu^+ \Delta E_\nu^{\max} + \tau_\nu^- \Delta E_\nu^{\min}, & \text{if } E_{\nu+1} > E_{\nu-1} \\ \tau_\nu^+ \Delta E_\nu^{\min} + \tau_\nu^- \Delta E_\nu^{\max}, & \text{if } E_{\nu+1} < E_{\nu-1} \end{cases}, \tag{1.6}$$

where τ_ν^+ and τ_ν^- are the forward and backward finite difference tangents, respectively (their definition is given in the following Section 1.2.2). The energy differences are given by

$$\Delta E^{\max} = \max\left(|E_{\nu+1} - E_{\nu}|, |E_{\nu-1} - E_{\nu}|\right),$$
$$\Delta E^{\min} = \min\left(|E_{\nu+1} - E_{\nu}|, |E_{\nu-1} - E_{\nu}|\right).$$

Now, through adequate choice of the initial path between the first and last image of the chain, one may calculate the MEPs of any transition containing a first order saddle point in the energy. Higher order saddle points have one or multiple degrees of freedom orthogonal to the path, in direction of which the energy can still be minimised. An educated guess or previous knowledge about a transition path will help in generating an initial path which quickly converges to the transition one wishes to study.

The initial path can be chosen arbitrarily and even randomly. Since the energy landscape of a nontrivial system is high-dimensional and complex, especially a random path can lead to unexpected local minima between the states and the discovery of new transition mechanisms. However, this is not a rigorous method of finding transition mechanisms and, in the author's experience, will only lead to the most obvious transitions.

Note that generating nontrivial initial paths for NEB calculations can be a time-consuming and difficult task. This is alleviated significantly by the graphical user interface (GUI) of *Spirit*, where the direct interaction with the system and a range of tools, designed to generate potentially complex states, enable one to quickly and effectively test and visually verify initial guesses. See Section 3.4 for more details.

To summarise the above, the algorithmic implementation of the NEB method is illustrated in Figure 1.3.

1.2.2 *Adaptation to spin systems*

So far, the definitions match those of the regular NEB method. However, for spin systems, special care has to be taken due to the fact that the phase space manifold $\mathcal{M}^{\text{phys}}$ is curved due to the constraint on the spin length, as noted in the Introduction, Section I.2 (see Equation (I.8)). The spins therefore move on unit spheres and the method is then called *geodesic* nudged elastic band (GNEB) method [56].

The expression for the overall distance $L_{\nu,\mu}$ is the norm of the vector of distances in the spin subspaces

$$L^{\nu\mu} = \sqrt{\sum_i (l_i^{\nu,\mu})^2}\,, \tag{1.7}$$

where $l_i^{\nu\mu}$ denotes the geodesic distance between the spin $\vec{n}_i^{\nu}$ at position i in image M^{ν} and the spin $\vec{n}_i^{\mu}$, also at position i, but in image M^{μ}. While in the Euclidean embedding space $\mathcal{E}$ the *geodesics* that measure the shortest distance between two points are straight lines, here they are great circles, and the distance $l_i^{\nu\mu}$ is thus given

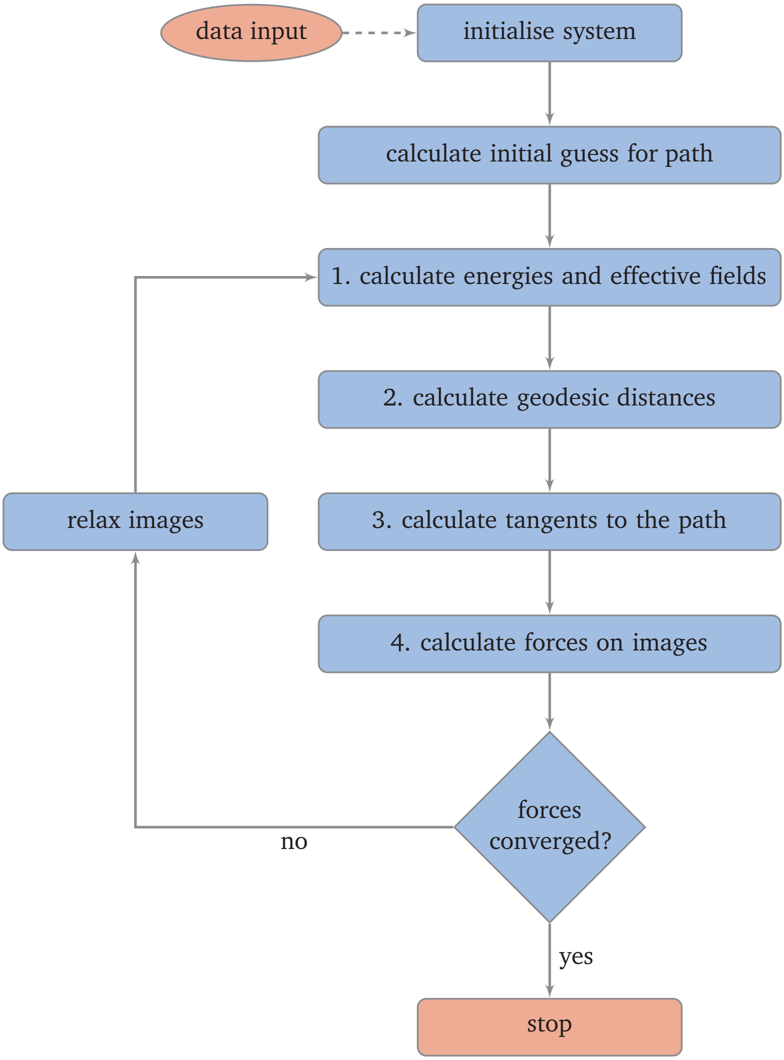

data input
initialise system
calculate initial guess for path
1. calculate energies and effective fields
2. calculate geodesic distances
relax images
3. calculate tangents to the path
4. calculate forces on images
forces converged?
no
yes
stop

FIGURE 1.3 – Flow chart of the GNEB algorithm

This flow chart visualises the general logic of the GNEB method. After initialising the system, an initial path has to be created. Subsequently, the iteration loop is started, in which the forces are calculated and, if they are not yet converged, applied. Image adapted from [108].

by

$$l_i^{\nu\mu} = \alpha_i^{\nu\mu} = \arccos\left(\frac{\vec{n}_i^{\nu} \cdot \vec{n}_i^{\mu}}{||\vec{n}_i^{\nu}||\ ||\vec{n}_i^{\mu}||}\right), \tag{1.8}$$

where $\alpha_i^{\nu\mu}$ is the angle between $\vec{n}_i^{\nu}$ and $\vec{n}_i^{\mu}$.

For spin systems we can still write simple forward and backward differences between images

$$\tau_\nu^+ = \frac{M_{\nu+1} - M_\nu}{|M_{\nu+1} - M_\nu|}, \quad \tau_\nu^- = \frac{M_\nu - M_{\nu-1}}{|M_\nu - M_{\nu-1}|}, \tag{1.9}$$

where the $M_\nu = \{\vec{n}_i\} \in \mathcal{M}^{\text{phys}}$ denote $3N$-dimensional vectors of the spin configurations. However, the tangents τ_ν, defined by Equation (1.5) and Equation (1.6), need to lie in the tangent space to their corresponding point M_ν on the manifold. One may correct the tangents for example by a simple projection, orthogonalizing the corresponding 3-component subvectors with respect to the spins

$$\vec{\tau}_{\nu,i} \rightarrow \vec{\tau}_{\nu,i} - (\vec{\tau}_{\nu,i} \cdot \vec{n}_{\nu,i})\vec{n}_{\nu,i}\,. \tag{1.10}$$

This tangent projection is illustrated for a single spin in Figure 1.4. Note that the tangents are required to be normalised (in $3N$) and therefore need to be re-normalized

$$\tau_\nu \rightarrow \tau_\nu / |\tau_\nu|\,. \tag{1.11}$$

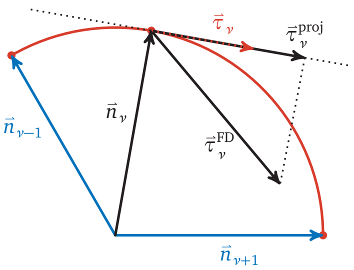


FIGURE 1.4 – Path tangents projection

Schematic visualisation of the projection of the tangents for a single-spin system. After a tangent τ_ν^{FD} is determined by finite difference calculation, it needs to be projected onto the tangent plane to the spin configuration so that it correctly points along the path. This tangent is denoted τ_ν^{proj} and can be calculated e.g. by removing the component in the direction of the image, see (1.10). Note that the tangent vector τ_ν needs to be normalized, which for a multi-spin system needs to be performed in $3N$ dimensions. Figure originally adapted from [108] and published in [47].

Finally, for the energy gradient force, the same scheme can be applied and we write for each spin

$$\vec{F}^{\text{E}}_{\nu,i} \rightarrow \vec{F}^{\text{E}}_{\nu,i} - (\vec{F}^{\text{E}}_{\nu,i} \cdot \vec{n}_{\nu,i})\vec{n}_{\nu,i}\,. \tag{1.12}$$

While being orthogonal to F_{S}, F_{E} also needs to be in the tangent plane, since spin length is conserved. The orthogonality constraint $F_{\mathrm{S}} \perp F_{\mathrm{E}}$ is applied so that the two forces do not interfere with each other, which would produce kinks in the path and in the worst case prevent convergence.

1.2.3 *Improvements on the regular NEB method*

Three notable improvements on the NEB method can be made:

- Climbing, falling and stationary images
- Spring force modulations
- Path shortening force

Without any addition to the NEB method, the probability of an image to lie precisely on the saddle point is virtually zero. Thus, we need to apply the so-called climbing image (CI) method, which will force the image with highest energy to move up to the saddle point, as illustrated in Figure 1.5. This is achieved through the deactivation of the spring force for that image, while applying the negative of the energy gradient along the path:

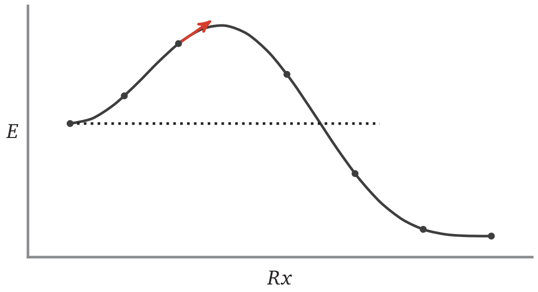


FIGURE 1.5 – Climbing image method

The CI method moves the image of highest energy towards the saddle point. This ensures that, when properly relaxed, the saddle point is known precisely. It is a critical tool for the GNEB method, as the saddle point can otherwise only be determined approximately. Image adapted from [108].

$$F_{\nu}^{\mathrm{S,CI}} = 0, \qquad F_{\nu}^{\mathrm{E,CI}} = -\nabla E_{\nu} + 2(\nabla E_{\nu} \cdot \tau_{\nu})\tau_{\nu}. \tag{1.13}$$

The alternative is to increase the number of images to attain a sufficient resolution, but this is neither computationally efficient, nor is it quantitatively guaranteed to be precise, which can be required, for example by the HTST method presented in the following Section 1.3. Note that in some cases, the CI can reveal the fact that a path is actually not a stable MEP, i.e. that the saddle point is not of first order, while the regular NEB would converge nonetheless.

Analogously, local minima can lie along the transition path and it can be very useful to converge a close-by image to the minimum. For this, one may apply a "falling image", simply by setting $F_{\mathrm{S}} = 0$ for the image, resulting in the image sliding down to the nearest local minimum due to the gradient force F^{E}_{ν}.

Note that in any case the images neighbouring a climbing (or falling) image will rearrange and the chain will effectively behave like two independent chains that meet at the climbing (or falling) image. As this behaviour can be useful in itself, the option to make images "stationary" has also been implemented in *Spirit*, enabling one in practice to take more precise influence on the convergence of a transition path.

The next improvement over the regular GNEB method is the equidistant placement of images along the energy curve. This can significantly improve the convergence of the CI onto the saddle point, as the resolution for the finite difference calculation of the tangents at the saddle point is increased. It is common to calculate cubic polynomials to interpolate between the discrete images on the path and generate a smooth energy curve $E(Rx)$ (see also Appendix A). The segment length of these polynomials can be calculated by numerical integration and the spring forces can be modulated between the images to produce equal distances along the interpolated curve. In *Spirit*, a weighting ratio between energy and reaction coordinate is implemented as a calculation parameter, with which one can vary the distribution of images along the reaction coordinate versus the energy.

A significant novelty which has been developed by the author and implemented in *Spirit* is the "path shortening force", which can be added in order to eliminate zero modes – degrees of freedom which do not change the energy – without a prohibitive amount of numerical overhead. Under certain conditions, the opportunity for an image to move along a zero mode can become a problem, hindering the convergence of the MEP. The concept presented is trivially simple: the path will be shortest when all tangents are parallel, i.e. the path is a straight line. Therefore, one can easily calculate finite differences to point a non-straight path towards the straight alignment by taking the finite difference of finite differences

$$F^{\mathrm{PS}}_{\nu} = \frac{M_{\nu+1} - M_{\nu}}{|M_{\nu+1} - M_{\nu}|} - \frac{M_{\nu-1} - M_{\nu}}{|M_{\nu-1} - M_{\nu}|}\,, \tag{1.14}$$

as visualised in Figure 1.6 for the case of a planar energy landscape. It is clear that the shortest trajectory is that which accumulates the least curvature.

It is clear that the path should only be optimised in this regard orthogonal to a) the energy gradient direction and b) the (normalised) path tangent:

$$F^{\mathrm{PS}}_{\nu} \;\rightarrow\; \underbrace{F^{\mathrm{PS}}_{\nu} - \left(F^{\mathrm{PS}}_{\nu} \cdot \frac{\nabla\mathcal{H}}{|\nabla\mathcal{H}|}\right)\frac{\nabla\mathcal{H}}{|\nabla\mathcal{H}|}}_{a)} \;\rightarrow\; \underbrace{F^{\mathrm{PS}}_{\nu} - (F^{\mathrm{PS}}_{\nu} \cdot \tau_{\nu})\tau_{\nu}}_{b)}\,. \tag{1.15}$$

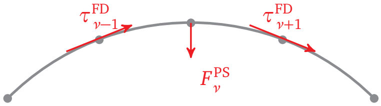


FIGURE 1.6 – Path shortening force to eliminate zero modes

The tangents are oriented towards growing reaction coordinate and the finite difference to correctly direct the path shortening force is therefore trivially $F^{\mathrm{PS}} = \tau^{\mathrm{FD}}_{\nu+1} - \tau^{\mathrm{FD}}_{\nu-1}$, where the τ^{FD}_{ν} are the normalised finite difference tangents (see Equation (1.14)). F^{PS}_{ν} points towards the shortest path between the neighbouring images $M_{\nu-1}$ and $M_{\nu+1}$. In order to work as intended, the path shortening force still needs to be orthogonalised to the gradient force and the tangent (tangents according to Equation (1.5) and Equation (1.6)) and scaled appropriately.

An additional step must be taken, as the above orthogonalisation in a sense still fails when the gradient orthogonal to the path is zero, which is exactly the case along a MEP. When an image is on the desired MEP, the path shortening force (1.14) will not be zero and may point to increasing energy, as the gradient force direction is undefined. The calculation would therefore not converge, but oscillate around the MEP. To solve this, the force can be normalised and scaled

$$F^{\mathrm{PS}}_{\nu} \rightarrow \max(|\nabla\mathcal{H}|, N\mathcal{F})\frac{F^{\mathrm{PS}}_{\nu}}{|F^{\mathrm{PS}}_{\nu}|}\,, \tag{1.16}$$

where N is the number of spins and $\mathcal{F}$ is the minimum value assigned to the force. This method is therefore to be applied as a kind of *pre-conditioner* for the regular GNEB method, applying the force with a minimum magnitude (set by the user), until the path is sufficiently converged so that zero modes should no longer cause it to malform. The minimum value for this force will prevent convergence to smaller values and the stronger it is, the more the path becomes distorted towards the homogeneous interpolation, i.e. the shortest possible path between the images. When the path is sufficiently pre-converged, the regular CI-GNEB method can be used to find the precise MEP.

Though one might expect that the curvature of the spin manifold $\mathcal{M}^{\mathrm{phys}}$ plays an important role in the finite differences between tangent vectors, it turns out that this approach works well.

1.2.4 *Test: Gaussian potentials*

Now that we have all the tools needed, we can apply the method to a simple test system, which is here chosen to be a single spin $\vec{n}$ in a gaussian superposition potential, defined as

$$\mathcal{H} = \sum_i \mathcal{H}_i = \sum_i a_i \exp\left(-\frac{l_i^2(\vec{n})}{2\sigma_i^2}\right)\,, \tag{1.17}$$

where a_i is the amplitude, σ_i the width and $l_i(\vec{n})$ is a measure of distance of the spin $\vec{n}$ to the center $\vec{c}_i$ of the i^{th} gaussian function. We now purposely choose *not* the geodesic distance between the vectors as the measure of distance, but instead

$$l_i(\vec{n}) = 1 - \vec{n} \cdot \vec{c}_i \,, \tag{1.18}$$

as the derivatives are easily calculated in this case (see Appendix B). Parameters for the test case are given in Table 1.1 and the script to produce the data using the Python API of *Spirit* is given in Appendix B.

a	σ	c_x	c_y	c_z
−1.10	0.06	−0.20	0.00	−0.90
0.80	0.15	−1.00	0.20	−0.20
−0.90	0.10	1.00	−0.20	−0.10
0.09	0.03	0.80	0.50	−0.80
0.15	0.07	0.80	−0.50	−0.70
−0.90	0.10	0.50	1.20	−0.40
−0.90	0.10	0.20	−0.90	−0.40

TABLE 1.1 – Parameters of Gaussian Hamiltonian

The parameters defined in this table provide for a sufficiently interesting energy landscape with several minima and saddle points (see Figure 1.7).

The example shown in Figure 1.7 concisely demonstrates how the climbing image (CI) addition to the GNEB method may help in obtaining the correct transition path without the need for an excessive amount of images. To illustrate the resulting paths, the energy barriers are plotted in Figure 1.8.

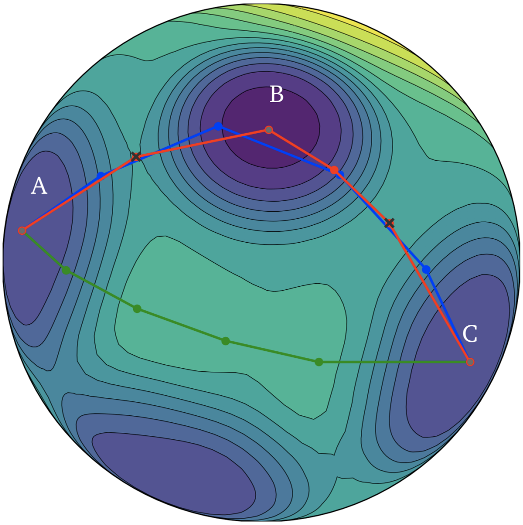


FIGURE 1.7 – Example of a GNEB calculation on a single spin

The energy landscape defined by Equation (1.17) and table 1.1 is plotted on a sphere, together with three transition paths: the great-circle interpolation between the two initial minima A and C (green); the minimum energy path calculated with the regular GNEB method (blue); the minimum energy path containing the exact saddle points (red, each saddle point marked by ×) and central minimum B, calculated by additionally imposing two climbing images (CIs) and the relaxation of the lowest-energy image. This image has been published in [47].

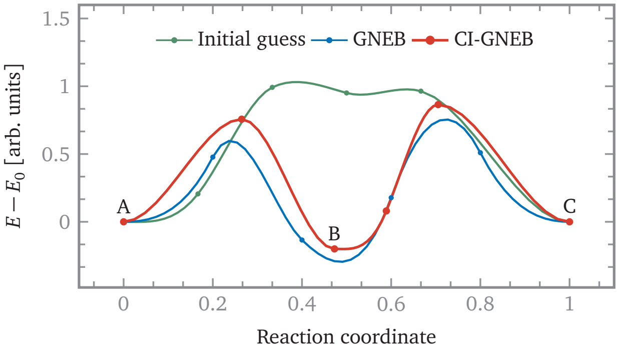


FIGURE 1.8 – Single spin energy barriers

The energy barriers corresponding to the paths shown in Figure 1.7. The great-circle interpolation between the initial and final images, A and C, gives an unnecessarily high barrier. The relaxation with the regular GNEB method gives an approximate MEP, where the minima and maxima along the path are not exactly known. The climbing image method and relaxation of the lowest-energy image produce the exact saddle points and minimum B along the minimum energy path. This graph has been published in [47].

With this very intuitive proof of concept, we can now turn to the more general case of systems of coupled spins. Monolayers containing skyrmions are a suitable test case due to the emergent complexity produced by the interactions between the spins.

Note that this simple example already demonstrates that in general, several initial guesses should be made, as well as some initial paths chosen randomly, in order to sufficiently sample the energy landscape for possible transitions. However, even with different initial guesses for the path, one could miss the fact that there are other surrounding minima and corresponding transitions, which the system could undergo. There is a need for a systematic method to seek out saddle points around a local minimum and Chapter 2 will elaborate the development of such a method and deal in more detail with this problem.

1.2.5 *Application to a 2D skyrmion texture*

In order to illustrate the usefulness for physical systems, the following shows the two known processes of skyrmion nucleation and annihilation: the radial collapse and the escape through a system boundary. We recall the Heisenberg Hamiltonian (I.1) from the introduction, here written with the contributing interactions:

$$\mathcal{H} = -\mu_S \sum_{i=1}^{N_S} \vec{B}^{\text{ext}} \cdot \vec{n}_i - \sum_{\langle ij \rangle} J_{ij} \vec{n}_i \cdot \vec{n}_j - \sum_{\langle ij \rangle} \vec{D}_{ij} \cdot (\vec{n}_i \times \vec{n}_j) \,.$$

Parameters are chosen as

$$\mu_S = 2\mu_B \,, \quad B^{\text{ext}} = 2.5\ \text{T} \,, \quad J_{ij} = 2\ \text{meV} \,, \quad D_{ij} = 0.6\ \text{meV} \,. \tag{1.19}$$

It is a point worth stressing again, exemplified by this kind of calculation, that *Spirit* can significantly improve scientific productivity by reducing the effort required for the combination of different methods, such as LLG and GNEB. This is illustrated by the following Python script

```
NOI=10
with state.State(input_file, quiet=True) as p_state:
    configuration.plus_z(p_state)
    chain.image_to_clipboard(p_state)
    chain.set_length(p_state, NOI)

    configuration.skyrmion(p_state, 5.0, phase=-90.0, idx_image=0)
    configuration.plus_z(p_state, idx_image=NOI-1)

    simulation.start(p_state, LLG, VP, idx_image=0)

    transition.homogeneous(p_state, 0, NOI-1)

    chain.update_data(p_state)
    parameters.gneb.set_image_type_automatically(p_state)
    simulation.start(p_state, GNEB, VP)
```

LISTING 1.1 –

Combination of LLG and GNEB to calculate the energy barrier and saddle point for the skyrmion collapse.

Within only a few lines of code, without intermediate steps or post-processing, this simple example can be run. Utilizing the powerful interactive features of the GUI, one can quickly reproduce the above (see also Section 3.4) and effectively deal with significantly more complex examples.

The minimum energy paths of both the collapse and the escape of the skyrmion through an open boundary of the system are shown in Figure 1.9.

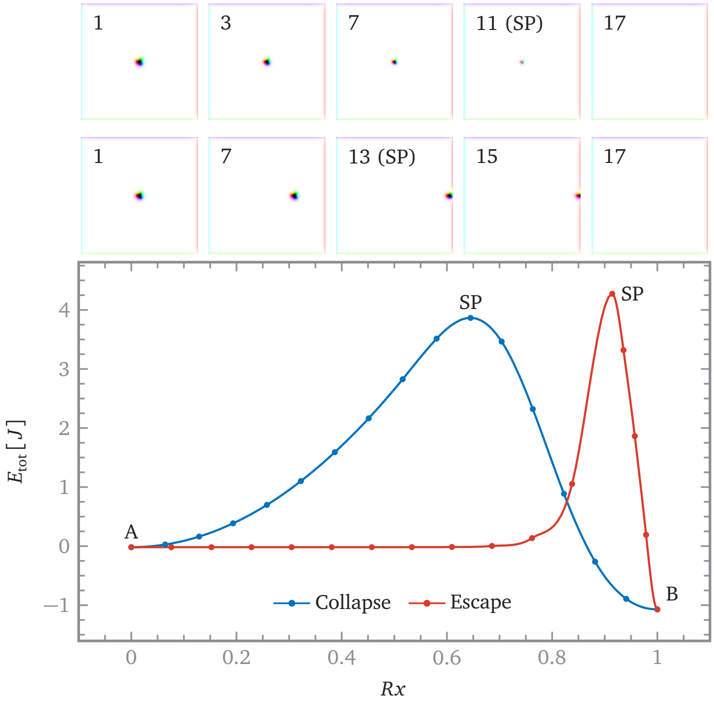


FIGURE 1.9 – Skyrmion transitions: minimum energy paths

Skyrmion collapse and escape transitions in a simple cubic monolayer of size $N \times N$. The barriers are of similar height, but different profile.

1.2.6 *Nontrivial 2D textures: skyrmion "sacks"*

To show the applicability of the GNEB method in more general cases, where more complicated transitions are possible, we will now look at a set of more general skyrmion states, recently described in Ref. [138]. We will see a large variety of transitions, which will need to be studied thoroughly if the stability of these objects is to be estimated. This will also showcase the usefulness of a software tool such as *Spirit* with a graphical user interface and advanced abilities to generate initial paths. We again use the Heisenberg Hamiltonian (I.1) from the Introduction, with parameters chosen to give $L_\mathrm{D} = 30a$ (for the parameter transformations see Equation (I.7)):

$$\mu_\mathrm{S} = 1\mu_B\,, \quad H^\mathrm{ext} = 0.65\,H_\mathrm{D}\,, \quad J_{ij} = 1\text{ meV}\,, \quad D_{ij} = 0.20944\text{ meV}\,. \tag{1.20}$$

For these parameters, skyrmionic sacks, or loops – also referred to as heavy skyrmions – can be stabilised and combined into new, arbitrarily large states. A few examples of these are shown in Figure 1.10.

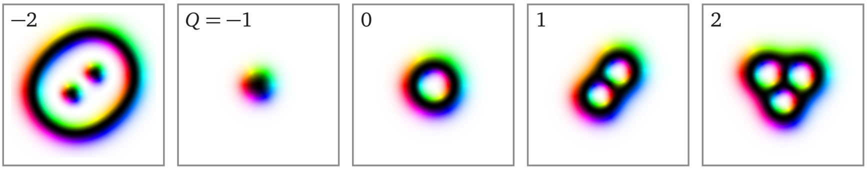


FIGURE 1.10 – Skyrmion "sacks"

Different skyrmionic configurations of varying topological charge Q (see also Equation (I.3)). The state $Q = -1$ is the regular skyrmion configuration. Notably, the skyrmionic ring has zero charge $Q = 0$, as it corresponds to a second skyrmion of opposite charge $Q = 1$ being inserted into the $Q = -1$ configuration. Therefore, the "sack" containing two regular skyrmions carries a charge $Q = -2$, while being a single, localised object. Inversely, by inserting further cores of charge $Q = 1$ into the $Q = 0$ configuration, one can create skyrmionic loops.

Note that the topological charge Q no longer uniquely defines a configuration, as becomes clear from the example of $Q = -2$, shown in Figure 1.10: the removal of one of the skyrmions on the inside will result in the same topological charge of $Q = -1$ as the regular skyrmion, but in a very different spin texture.

The $k\pi$ skyrmion [139], resembling k nested domain wall rings, will in the case of $k = 2$ and $Q = 0$ be referred to as "skyrmionium" [140, 141]. In contrast, the alternative higher order combination of domain wall rings, such as the $Q = 1$ and $Q = 2$ examples shown in Figure 1.10, will be referred to as skyrmionic "loops". Far more complicated states can be created, as shown in Ref. [138], but we will see in the following that these few states already pose significant complexity and the GNEB calculations require effort.

Having stabilised these objects, we can now show that the notion of topological charge is unsuitable to characterise these configurations. It should be clear that the $Q = 0$ skyrmionium configuration can be destroyed by a continuous transformation of the vector field, i.e. by a continuous homotopy transformation, such as visualised in Figure 1.11.

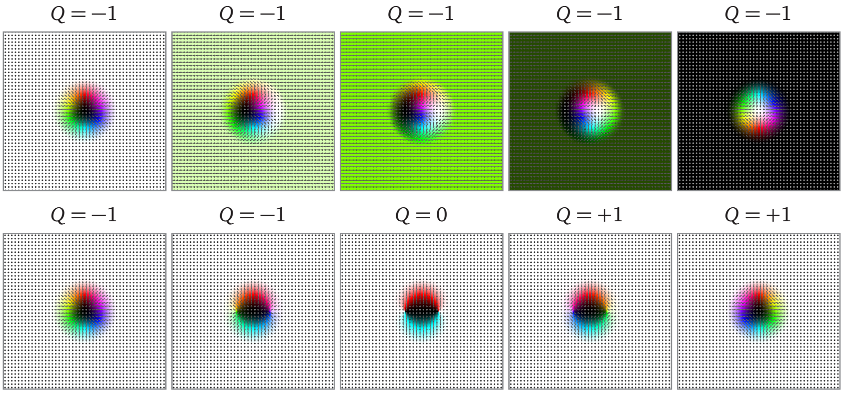


FIGURE 1.11 – Transformations between a skyrmion and an antiskyrmion

Top: homotopy transformation between a skyrmion and antiskyrmion of topological charge $Q=-1$, induced by rotation of all spins by 180 deg around the y-axis. Bottom: non-homotopy transformation between a skyrmion with $Q=-1$ and an antiskyrmion with $Q=1$, induced by linear interpolation between the two states, where two defects appear along the path. The latter transformation cannot be achieved without two defects, as the topological charge of the state needs to be incremented twice.

The shown transformations illustrate the necessity of nucleating two defects in order to change the topological charge from $Q=-1$ to $Q=1$, which is analogous to what is observed in the switching of skyrmions by external field pulses [142]. Note that the transformation between two skyrmions of Néel and Bloch chirality is also a homotopy transformation, corresponding simply to the rotation of all spins around the z-axis. While the transformations between skyrmions are quite illustrative, a more formal visualisation of a homotopy transformation is given in Figure 1.12.

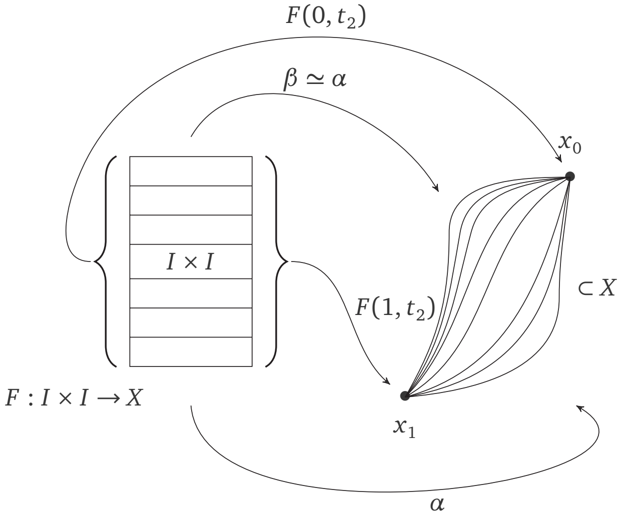


FIGURE 1.12 – Homotopy transformation

Roughly speaking, two functions are called homotopic when the can be continuously deformed into one another – said deformation being a homotopy transformation. This is illustrated here by the transformation $F : I \times I \to X$ between topological spaces of the flat grid $I \times I$ to a warped set of lines connecting the points $x_0 = F(0,t_2)$ and $x_1 = F(1,t_2)$.[1] The continuity of such transformations implies that no "knots" can be made or untied by a homotopy transformation, i.e. topological winding numbers are invariant under such transformations.[2] Consequently, in the continuum limit, a skyrmion in a homogeneous background cannot be destroyed by such a transformation. The skyrmionium with $Q = 0$, however, can collapse its ring shape without any discontinuity. Image recreated from [143] by Stefan Kottwitz and Alain Matthes.[3]

From this it should be clear that between the different variations of states with the same topological charge Q there is always a homotopy transformation, however potentially corresponding to a significant change in the configuration of the system. The $Q = 0$ skyrmionium is topologically equivalent to the homogeneous ferromagnetic state with the same topological charge and a homotopy transition between the two exists, as shown by example in Figure 1.13.

[1] Note that this is not a Homeomorphism

[2] The concept is illustrated nicely by the transformation of a cup into a torus (see `https://wikipedia.org/wiki/Homotopy`)

[3] See also `http://www.texample.net/tikz/examples/homotopy/`

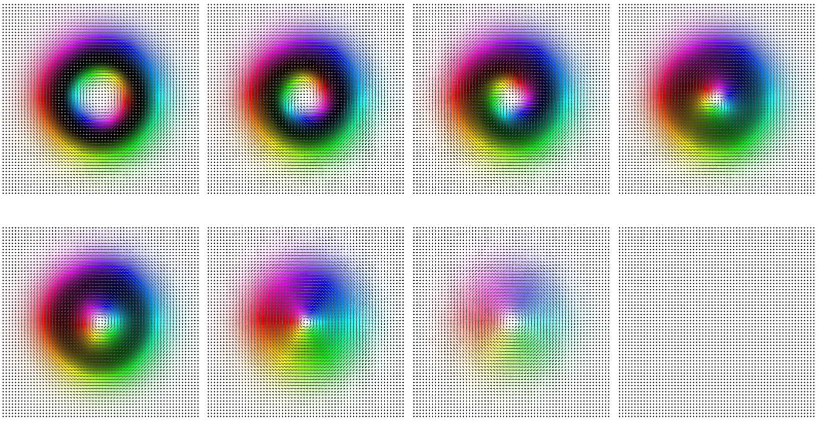


FIGURE 1.13 – Homotopy transformation of the skyrmionium

A topologically trivial transformation which destroys the skyrmionium. In the atomistic model, it is common for this kind of transition to be energetically more costly than the nucleation of a defect, such as shown in the following MEP calculations in this section.

Attempts to stabilise a minimum energy path (MEP) corresponding to a homotopy transition between the skyrmionium and the ferromagnetic state were not successful – not even with the help of the GUI tools (see also Section 3.4) – and the nucleation of one or multiple defects appeared instead. The homotopy transformations which can destroy the $Q = 0$ state are not a trivial transition of the system and are accompanied by a significant change in energy.

This means that the transitions and lifetimes of magnetic configurations cannot in general be described by topology, but are instead determined entirely by the energy landscape. In fact, while minimum energy paths (MEPs) may include topological transitions, they do not have to, and it is important to note that therefore the saddle point of the transition, too, need not coincide with a topological transition. This is a signature of the discrete lattice model, as in continuum theory, topological transitions are always accompanied by energy divergencies, meaning they represent saddle points of the energy landscape. The fact that the continuum model is indeed insufficient for the estimation of the thermal stability of topological states can be further underlined by HTST calculations, which show nontrivial dependencies, not only on the energy barriers, but also on entropic contributions [40, 144] and the underlying lattice geometry [145]. These insights warrant some further investigation of these higher order skyrmionic states using the GNEB and HTST methods. Note that some calculations have already been made, where in fact *Spirit* was used to perform GNEB calculations [139].

Due to computational time constraints, the GNEB calculations in this section are presented for $L_D = 30$ and a lattice size limited to 130 × 130, which is too small to achieve quantitative precision for some of the larger states. However, the qualitative

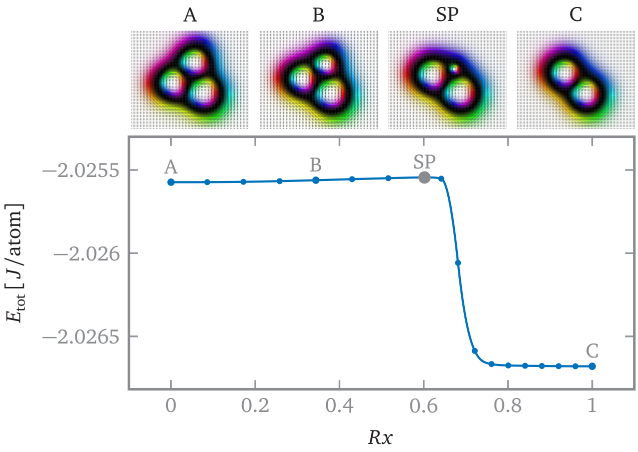


FIGURE 1.14 – Skyrmionic core collapse inside a $Q = 2$ skyrmion "sack"

MEP calculation for the collapse of a $Q = 1$ core inside the skyrmionic $Q = 2$ state. Note that the collision transition of two cores is not a stable MEP inside of this configuration. The energy barrier is in this case below the precision of the calculations, which is limited by the system size at a value of $\Delta E \approx 0.3$ meV, but in any case an order of magnitude smaller than the energy difference of $\Delta E = 11.1$ meV between the $Q = 2$ and $Q = 1$ states.

relations between the energy barriers for the different transitions hold.

Figure 1.14 shows the MEP of the collapse of one of the skyrmionic cores inside a skyrmionic loop of charge $Q = 2$, resulting in a loop with $Q = 1$.

As can be seen in Figure 1.14, the barrier for the collapse of one of the cores is an order of magnitude smaller than the energy difference between the two loops with $Q = 2$ and $Q = 1$ respectively.

This kind of transition path, where only a portion of a localised state is changed, is difficult to create automatically and can even pose some challenges when using the direct interaction tools of the GUI of *Spirit* (see also Section 3.4). However, together with the real-time visualisation of the images on the transition path, the results of actions can be checked and paths can be manually corrected until they converge as desired. It should be noted that the spring force modulations, which redistribute the images along the interpolated energy curve instead of just along the reaction coordinate (see Section 1.2.3), and the path shortening force (see Figure 1.6), which were both added to *Spirit* after these calculations were performed, significantly improved the convergence of paths such as shown here, where sharp rises in the energy appear close to the saddle point.

The collapse of a skyrmionic core in such a loop configuration is only one of the possible transition. As will be detailed in Chapter 2, during the course of this work a transition corresponding to the collision of two skyrmions was found to be a stable minimum energy path. Subsequently, this transition was also tested in this system and found to result in a stable MEP, as shown in Figure 1.15 in the loop with $Q = 1$, resulting in a transition to the skyrmionium ($Q = 0$).

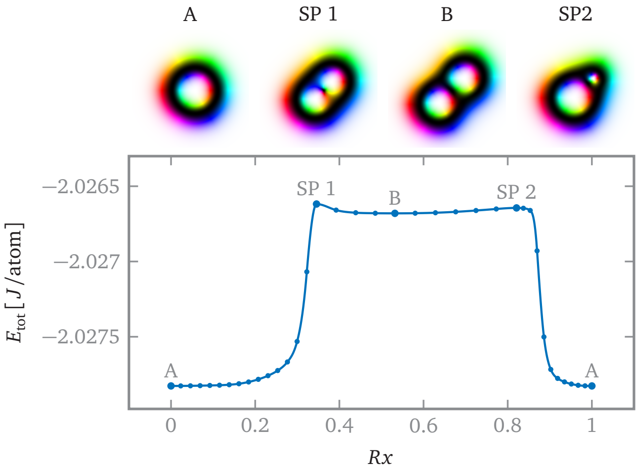


FIGURE 1.15 – Two transitions between $Q = 1$ and $Q = 0$ skyrmion "sacks"

MEP calculation for the transitions between a skyrmionic $Q = 1$ state and a $Q = 0$ skyrmionium state. Note that the collision transition of two skyrmions (see transition A – SP1 – B) has only recently been initially reported by the author of this work [58]. The energy difference between the two states is, similar to the case in Figure 1.14, $\Delta E \approx 11.5$ meV and the collapse of one of the cores has again a barrier of $\Delta E \approx 0.3$ meV. The energy barrier of the collision is on the same order of magnitude as the collapse at $\Delta E \approx 0.6$ meV .

It is evident that the behaviour is generally the same as in the previous case, but while the collision between the two skyrmionic cores has a higher energy barrier than the collapse of one of the cores, the energy difference between the states of $Q = 2$ and $Q = 1$ is still on a different order of magnitude. Qualitatively speaking, the skyrmionic loop seems to be a tightly wound structure, which seems to cost a relatively high amount of energy, while its tendency to shrink in size pushes the cores inside so close together that they are easily collapsed or collided. This also holds for the collapse of the skyrmionic core inside the $Q = 0$ skyrmionium, as shown in Figure 1.16.

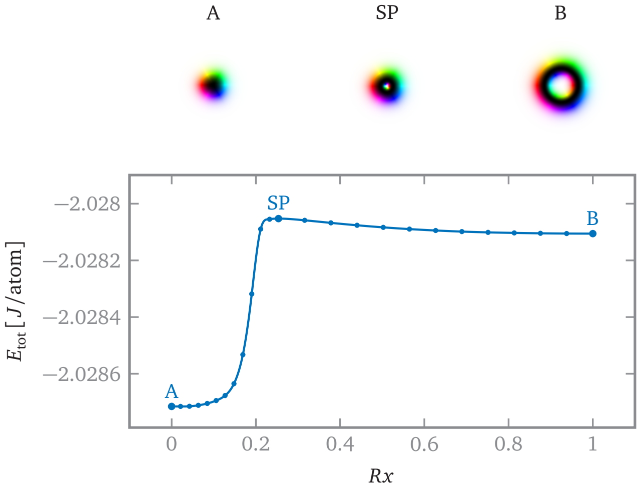


FIGURE 1.16 – Collapse of a skyrmionium to a skyrmion

MEP calculation for the transition from a skyrmion with $Q = -1$ state to a $Q = 0$ skyrmionium – or inversely the collapse of the skyrmionium. The energy difference between the two states is $\Delta E_{\text{A}-\text{B}} \approx 10.3$ meV, while the energy barrier of the collapse is $\Delta E_{\text{B}-\text{SP}} \approx 2$ meV.

It appears that such skyrmionic loops are not particularly stable, but this should be studied under different parameters as well. As the energy barriers do not change significantly with the charge Q, the calculation of barriers for loops of low charge would already provide a lot of information in general.

We will now examine the skyrmion sacks, such as the $Q = -2$ state shown in Figure 1.10 which behave quite differently from the previous loop configurations. Due to the fact that the energy difference between the states of different topological charge is significantly lower in this case (see Ref. [138]), the barriers can be expected to be – relatively – higher. The first example of this is shown in Figure 1.17, where two transitions between the sacks of $Q = -1$ and $Q = -2$ are shown.

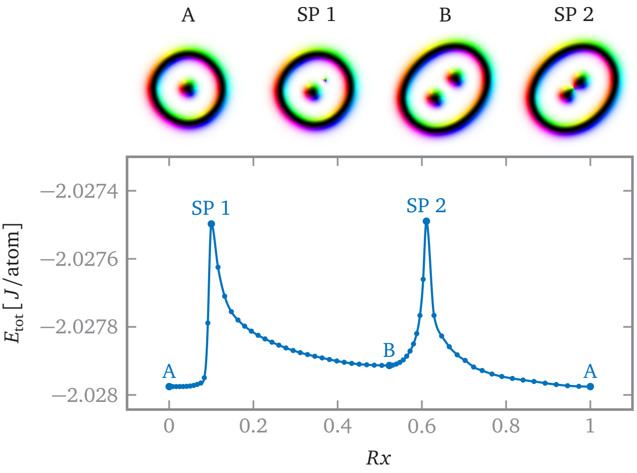


FIGURE 1.17 – Two transitions between $Q = -2$ and $Q = -1$ skyrmion "sacks"

MEP calculation for the transitions between a skyrmionic $Q = -1$ state and a $Q = -2$ state. Note the analogous collision transition as in Figure 1.15 (see transition B – SP1 – A) recently first reported by the author of this work [58]. The energy difference between the two states is $\Delta E_{\text{A-B}} \approx 1$ meV, while the energy barrier of the collision is $\Delta E_{\text{B-SP2}} \approx 7$ meV and the collapse of one of the cores has a barrier of $\Delta E_{\text{B-SP1}} \approx 7$ meV.

The conjecture is found to be correct, as the barriers are now almost an order of magnitude larger than the energy difference between the states – the relation has been almost inverted. The distance between the skyrmion cores is significantly larger inside the sack than in the case of the skyrmionic loops, as the energy cost for the deformation and size increase of the surrounding wall is observed to be very small. This effectively puts far less pressure on the skyrmions and there is consequently a higher barrier for their collapse or collision.

Equivalent transitions between sacks with $Q = -3$ and $Q = -2$ are presented in Figure 1.18.

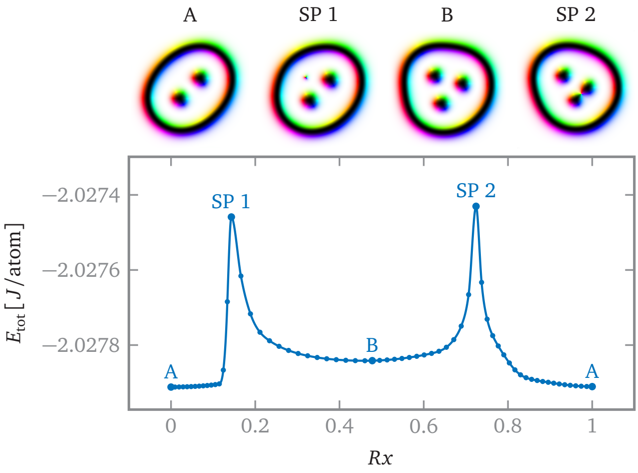


FIGURE 1.18 – Two transitions between $Q = -3$ and $Q = -2$ skyrmion "sacks"

MEP calculation for the transitions between a skyrmionic $Q = -2$ state and a $Q = -3$ state. The energy difference between the two states is $\Delta E_{\mathrm{A-B}} \approx 1$ meV, while the energy barrier of the collision is $\Delta E_{\mathrm{B-SP2}} \approx 7$ meV and the collapse of one of the cores has a barrier of $\Delta E_{\mathrm{B-SP1}} \approx 6.5$ meV.

Clearly, the tendency is the same and the relative energy differences can be expected to behave the same for arbitrarily large skyrmion sacks.

One transition remains to be shown: the collision of a skyrmion with the boundary wall of the sack. This is shown for the sacks of $Q = -3$ and $Q = -2$ in Figure 1.19. Note that this transition was not found to be a MEP for the transition from $Q = -1$ to $Q = 0$.

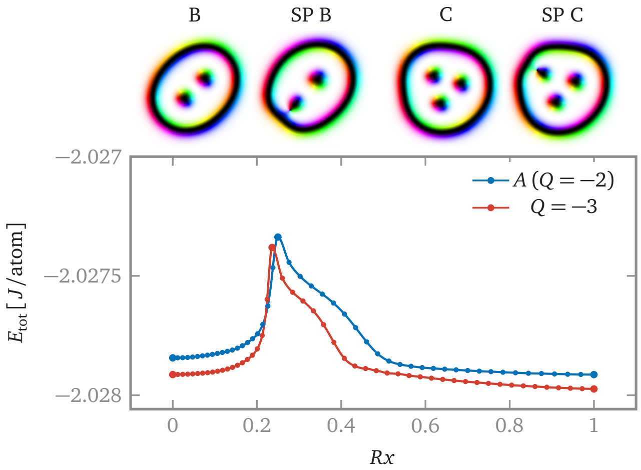


FIGURE 1.19 – Skyrmion colliding with the "sack" wall

The energy barriers are equal within the precision of these calculations at $\Delta E \approx$ 9 meV. The skyrmionium does not support the collision of the core and boundary as a first-order saddle point.

Once more, the energy barrier for the transition is an order of magnitude larger than the energy difference between the topologically distinct states.

Finally, there is a transition which is highly important for the stability of these objects: the breaking of the sack. This transition was not found for the skyrmionium, but is shown for charges $Q = -1$ to $Q = -3$ in Figure 1.20.

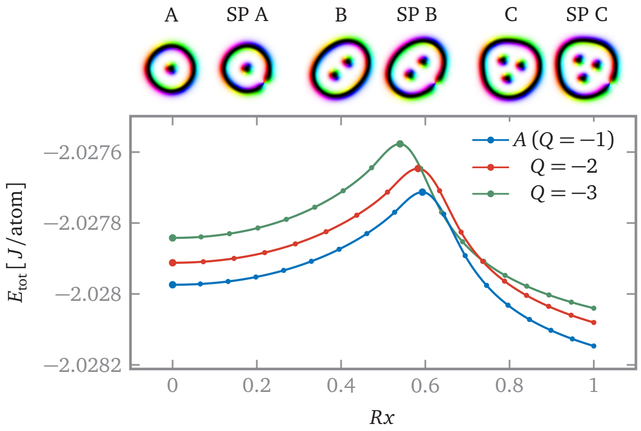


FIGURE 1.20 – Breaking the outer wall of a skyrmion "sack"

The curves have slightly different shapes due to the fact that their boundary images for the broken sack is slightly different. The energy barriers are approximately equal at $\Delta E \approx 4$ meV. Note that the MEP does not constitute a meaningful transition after the saddle point, as the state is broken and no useful information is given by the rest of the transition.

Note that the position of the defect nucleation along the wall can change the energy barrier, though not to a significant extent. However, if one were to attempt transition rate calculations (see the following Section 1.3), this would be a difficult issue to resolve correctly, especially as *counting* the number of saddle points and their corresponding transition rates is an important part of the calculations.

It has been shown that a wide variety of interesting transitions can appear in chiral magnetic systems, but aside from the energy barrier there is not yet any quantitative measure of the likelihood of a transition actually occurring. The following Section 1.3 will introduce the mathematical tools to calculate approximate transition rates out of the properties of the energy landscape, instead of having to rely on statistical sampling methods.

1.3 TRANSITION RATE CALCULATIONS

In the estimation of the lifetimes of magnetic states, the time-scales between the fast precession of magnetic moments relative to the transitions between states, may prohibit the use of dynamical simulations (see also Section I.2 and Section I.4). Therefore, statistical approaches to the calculation of transition rates may be preferable [57]. The calculation of energy barriers with GNEB can give a first impression of the stability of a state with respect to specific decay processes, but as a quantitative measure it is highly unreliable [38, 40]. This section will deal with HTST as a method to retrieve a more suitable measure of a states lifetime – a transition rate following an Arrhenius-type law

$$\Gamma = \Gamma_0 e^{-\Delta E/k_\mathrm{B}T}\,, \tag{1.21}$$

where ΔE is the energy barrier for a given transition and the prefactor Γ_0, corresponding to a fundamental fluctuation rate expressing the characteristic time scales of the dynamics of the barrier-crossing, is determined by the properties of the energy landscape. The discussion is kept close to references [57] and [146].

1.3.1 *Transition state theory*

Transition state theory (TST) [147] has been widely used to estimate the rate of thermally activated atomic rearrangements, such as chemical reactions and diffusion [148]. The separation of time scale mentioned above makes it possible to estimate the rate from the probability of finding the system in the most restrictive and least likely *region of phase space*, separating the initial state from possible final states – the *transition state*. A general expression for the rate of escape from an initial state is [149]

$$\Gamma = \langle \delta[f(\mathbf{x})] v_\perp(\mathbf{x}) \chi[\eta(t)] \rangle\,, \tag{1.22}$$

where $\mathbf{x}$ represents the degrees of freedom of the system, angled brackets denote thermal averaging with a Boltzmann distribution, $f(\mathbf{x}) = 0$ defines a dividing surface, which separates the initial state from other stable configurations in the remaining configuration space, $\delta[f(\mathbf{x})]$ restricts the averaging to the dividing surface, $v_\perp(\mathbf{x}) = \nabla f(\mathbf{x})\dot{\mathbf{x}}$ is the projection of the velocity on the local normal vector of the dividing surface, and finally $\chi[\eta(t)]$ is the functional of a full trajectory described by $\eta(t)$.

The functional $\chi[\eta(t)]$ includes coupling to the heat bath and takes the value of unity if a trajectory, starting from the initial state, goes directly from a point $\mathbf{x}$ on the dividing surface to the final state and remains there for a long time compared to the time it takes to cross the barrier, but is zero otherwise. In the transition state theory (TST) approximation, all trajectories pointing away from the initial state at the dividing surface are assumed to be reactive, i.e. the functional $\chi[\eta(t)]$ is approximated by a Heaviside step function $h[v_\perp(\mathbf{x})]$. This means that recrossings of the dividing surface are neglected in the TST. Calculations of short time dynamical trajectories can then be carried out starting at the dividing surface to determine the recrossing correction factor [150, 151], κ, and obtain the exact rate $\Gamma = \kappa\Gamma^{\mathrm{TST}}$. Note, however, that recrossing corrections for spin systems have not yet been derived.

Given a transition state dividing surface, $f(\mathbf{x}) = 0$, the reaction rate constant Γ^{TST}, corresponding to the flux through the dividing surface, can be estimated for spin systems, in a way that is analogous to atomic systems [152, 153], as

$$\Gamma^{\mathrm{TST}} = \frac{1}{C} \int_R \underbrace{e^{-E(\mathbf{x})/k_\mathrm{B}T}}_{\text{a)}} \underbrace{v_\perp(\mathbf{x})}_{\text{b)}} \underbrace{\delta[f(\mathbf{x})]}_{\text{c)}} \underbrace{h[v_\perp(\mathbf{x})]}_{\text{d)}} \, d\mathbf{x}, \tag{1.23}$$

where R denotes the region associated with the initial state up to and including the dividing surface, the normalization constant – assuming a Boltzmann distribution $dP = C\exp(-\frac{E}{k_\mathrm{B}T})d\mathbf{x}$ around the local minimum – is given by

$$C = \int_R \underbrace{e^{-E(\mathbf{x})/k_\mathrm{B}T}}_{\text{a}} d\mathbf{x} \tag{1.24}$$

and

a) is the Boltzmann distribution,

b) is the velocity along the dividing surface normal: $v_\perp(\mathbf{x}) = \nabla f(\mathbf{x})\dot{\mathbf{x}}$,

c) ensures that only the dividing surface is taken into account,

d) is the Heaviside function to eliminate recrossing.

The dividing surface $f(x)$ can be chosen arbitrarily, as long as it contains the saddle point and its normal is parallel to the unstable mode in this point. For convenience, we choose a hyperplane, orthogonal to the unstable mode. Note, the length of the velocity vector $v_\perp(\mathbf{x})$ is proportional to the magnitude of the energy gradient but due to condition c), $\delta[f(\mathbf{x})]$, for our choice of dividing surface, the damping contribution to the velocity points inside the hyperplane, while the precessional contribution points orthogonal to it. This is due to the fact that any point on the hyperplane, outside of the saddle point, naturally has a gradient pointing inside the plane towards the saddle point.

1.3.2 *HTST: harmonic approximation*

The transition rate (1.23) can be derived analytically in the harmonic approximation to TST [155]. We expand the energy around a spin configuration $M_0 = \{\vec{n}_{0,i}\} \in \mathcal{M}$ (see also Equation (I.8)) and make the harmonic approximation

$$E \approx E_0 + G_i n_i + \frac{1}{2} n_i H_{ij} n_j, \tag{1.25}$$

where G is the covariant gradient and H the covariant Hessian matrix of second derivatives at M_0. The corresponding energy landscape contributions to Equation (1.23) are schematically visualised in Figure 1.22.

While Section 2.2 of the following chapter will cover the derivation of the covariant Hessian matrix at nonstationary points, the following HTST calculations only require

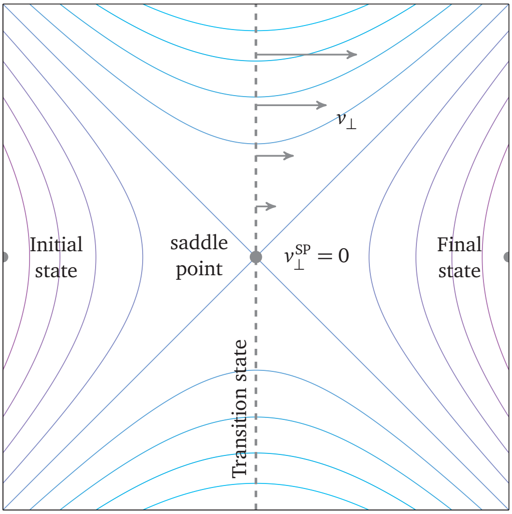


FIGURE 1.21 – Visualisation of the dividing surface in HTST

Schematic visualisation of a region around a first order saddle point on an energy surface for a magnetic spin system. The hyperplanar transition state (also called dividing surface), used in HTST, is shown as a dashed line. It includes the saddle point and its normal vector points along the unstable mode. The velocity orthogonal to the dividing surface vanishes at the saddle point according to the LLG equation of motion. Reprinted from [154] with minor modifications, licensed under CC BY-NC-ND 4.0.

the Hessian at local minima and first order saddle points of the energy landscape, where it can be derived more easily (see Appendix C). Note that $G = 0$ at stationary points of the energy landscape. Note that, writing

$$M = M_0 + q\,, \quad |q| \ll 1\,, \tag{1.26}$$

the above formulation is possible due to $G_i n_{0,i} = 0$ and $H_{ij} n_{0,j} = 0$, as the covariant gradient and Hessian – G and H – live in the tangent space to the physical manifold $\mathcal{M}$ at M_0 (see also sections I.2 and 2.2).

We choose the perturbations q_i, which are required to be tangential to M_0, to lie along the eigenvectors Λ_i of the Hessian, i.e. $q = \sum q_i \Lambda_i$, as visualised schematically in Figure 1.22. This yields us

$$E^{\mathrm{HTST}} = E_0 + \frac{1}{2} \sum_i^{2N} \lambda_i q_i^2\,, \tag{1.27}$$

where λ_i are the eigenvalues of the Hessian corresponding to the $2N$ tangential eigenvectors Λ_i. We assume an ordering of the eigenmodes by their eigenvalue, meaning that the unstable mode at the saddle point is Λ_1^{SP}.

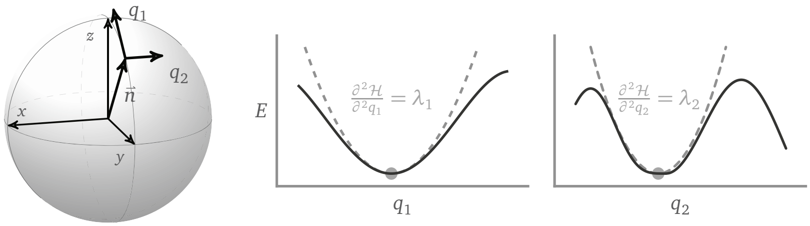

q_1
z
q_2
$\vec{n}$
x
y
E
$\frac{\partial^2\mathcal{H}}{\partial^2 q_1} = \lambda_1$
$\frac{\partial^2\mathcal{H}}{\partial^2 q_2} = \lambda_2$
q_1
q_2

FIGURE 1.22 – Schematic of spin perturbation

Illustration of the tangential perturbations to a single spin. Left: the spherical manifold of the spin vector $\vec{n}$ and tangential perturbations q_1 and q_2. Right: the harmonic approximation to the energy landscape assumes a parabolic potential close to the point of expansion. The eigenmode directions have different curvatures of the energy, given by the Hessians eigenvalues λ_1 and λ_2. Note that the eigenmode directions can point in arbitrary tangential directions and in the case of multiple spins need not be orthogonal for each spin, but only in total.

In this notation of perturbations, the LLG equation is linearized to the dynamical equation

$$\partial_t \vec{n}_i = \mathcal{V}_{ij} \vec{n}_j \,, \tag{1.28}$$

where $\mathcal{V}$ is generally called the dynamical matrix, but in the context of the LLG equation sometimes velocity matrix. Of course $\partial_t \vec{n}_{0,i} = \mathcal{V}_{ij} n_{0,j} = 0$. This lets us rewrite the transition rate Γ^{TST} of Equation (1.23) into its harmonic approximation

$$\begin{aligned} \Gamma^{\mathrm{HTST}} &= \frac{1}{C} \int_R \exp\left(-\frac{1}{k_{\mathrm{B}} T} E(\mathbf{x})\right) v_\perp(\mathbf{x})\, \delta[f(\mathbf{x})] h[v_\perp(\mathbf{x})]\, d\mathbf{x}_i \\ &= \frac{1}{C} \int_V \exp\left(-\frac{1}{2k_{\mathrm{B}} T} \sum_{i=2}^{2N} \lambda_i^{\mathrm{SP}} q_i^2\right) \mathbf{a} \cdot \mathbf{q} \prod_{l=2}^{2N} dq_l \; e^{-E^{\mathrm{SP}}/k_{\mathrm{B}} T} \,. \end{aligned} \tag{1.29}$$

The energy $E(\mathbf{x})$ is expanded quadratically around the saddle point according to Equation (1.27), where the E_0 contribution gives the final exponential. The Heaviside function $h[v_\perp(\mathbf{x})]$ results in the requirement $v_\perp > 0$. The delta function $\delta[f(\mathbf{x})]$ restricts the integration to the dividing surface $f(\mathbf{x})$ – the hyperplane to which the negative mode is orthogonal – so that the sum and product start at $i = 2$. The perpendicular velocity $v_\perp$, which points along the negative mode, is rewritten by representing the velocities $v = \partial_t \vec{n} = \dot{q}$ of the dynamical Equation (1.28) in the Hessian matrix' eigenbasis, writing

$$v_\perp = \dot{q}_1 = \mathcal{V}_{1j}^{\mathrm{SP,P}} q_j := \mathbf{a} \cdot \mathbf{q} \,, \tag{1.30}$$

where $\mathcal{V}^{\mathrm{SP,P}}$ is the precessional part of the dynamical matrix at the saddle point represented in the $2N$ tangent space (i.e. the same space as q)

$$\mathcal{V}|^{2N} = \Lambda^T \mathcal{V}|^{3N} \Lambda \,. \tag{1.31}$$

As the damping contributions to $\dot{q}_1$ on the dividing surface necessarily would point inside the dividing surface (towards the saddle point), they are zero. Therefore, $\vec{a}$ is the first line of the dynamical matrix $\mathcal{V}$ represented in the ($2N$-) basis of tangential perturbations (by our choice the Hessians eigenbasis). Note that the dynamical matrix (1.31) can also be represented differently, by using Equation (I.10).

At this stage we factor modes of zero eigenvalue, also called zero modes or Goldstone modes, out of the integral. As the corresponding exponentials in Equation (1.29) will take the value $e^{\lambda_i q_i^2} = e^0 = 1$ and the corresponding velocity factors are $a_i = 0$, we can pull the integration over those q_i out of the product of integrals as a factor

$$V^{\mathrm{SP}} = \int \prod_{i=2}^{N_0^{\mathrm{SP}}} dq_i \,. \tag{1.32}$$

The following derivation has been published in different form [155], but the comparatively short version presented here was created from notes provided by Pavel Bessarab and Igor Lobanov. In order to further simplify the integral, we rescale the remaining integrals:

$$x_i = q_i \sqrt{\frac{\lambda_i}{2k_{\mathrm{B}}T}}\,, \tag{1.33}$$

$$b_i = a_i \sqrt{\frac{2k_{\mathrm{B}}T}{\lambda_i}}\,, \tag{1.34}$$

$$\Gamma^{\mathrm{HTST}} = \frac{V^{\mathrm{SP}}}{C} \frac{\sqrt{2k_{\mathrm{B}}T}^{P^{\mathrm{SP}}}}{\prod_i' \sqrt{\lambda_i^{\mathrm{SP}}}} \int_V \exp\left(-\sum_i{}' x_i^2\right) \mathbf{b}\cdot\mathbf{x} \prod_l{}' dx_l \; e^{-E^{\mathrm{SP}}/k_{\mathrm{B}}T}\,, \tag{1.35}$$

where $P^{\mathrm{SP}} = 2N-(1+N_0^{\mathrm{SP}})$ is the number of positive modes (i.e. $\lambda_i > 0$) at the saddle point and $\prod'$ and $\sum'$ denote indexing only over positive eigenvalues, i.e. in this case over $[2+N_0^{\mathrm{SP}}, 2N]$. We apply the condition that $v_\perp = \mathbf{a}\cdot\mathbf{q} = \mathbf{b}\cdot\mathbf{x} > 0$ (elimination of recrossings by the Heaviside function $h[v_\perp(\mathbf{x}]$) and make use of the invariance of the integral under rotation of the tangential basis, writing

$$\mathbf{b}\cdot\mathbf{x} \overset{!}{=} |b| x_p \;\Rightarrow\; x_p > 0\,, \tag{1.36}$$

where Equation (1.34) trivially gives

$$|b| = \sqrt{2k_{\mathrm{B}}T}\sqrt{\sum_i{}' \frac{a_i^2}{\lambda_i^{\mathrm{SP}}}}\,. \tag{1.37}$$

This gives us separation of variables in the integral so that the Equation (1.35) reduces to the evaluation of the Gaussian integrals

$$\int\limits_{-\infty}^{\infty} e^{-x^2} dx = \sqrt{\pi} \quad \text{and} \quad \int\limits_{0}^{\infty} x e^{-x^2} dx = \frac{1}{2},$$

$$\begin{aligned} \Gamma^{\mathrm{HTST}} &= \frac{V^{\mathrm{SP}}}{C} \frac{\sqrt{2k_\mathrm{B}T}^{\,P^{\mathrm{SP}}}}{\prod_i' \sqrt{\lambda_i^{\mathrm{SP}}}} |b| \int\limits_{-\infty}^{\infty} \prod_{l \neq p}{}' dx_l \int\limits_{0}^{\infty} dx_p x_p \prod_i{}' e^{-x_i^2} \, e^{-E^{\mathrm{SP}}/k_\mathrm{B}T} \\ &= \frac{1}{2\pi} \frac{V^{\mathrm{SP}}}{C} \frac{\sqrt{2\pi k_\mathrm{B}T}^{\,P^{\mathrm{SP}}+1}}{\prod_i' \sqrt{\lambda_i^{\mathrm{SP}}}} \sqrt{\sum_i{}' \frac{a_i^2}{\lambda_i^{\mathrm{SP}}}} \, e^{-E^{\mathrm{SP}}/k_\mathrm{B}T} \,. \end{aligned} \tag{1.38}$$

We apply the same process to retrieve the normalisation constant (note in this case there is no negative mode):

$$\begin{aligned} C &= \int_R \exp\left(-\frac{1}{k_\mathrm{B}T} E(\mathbf{x})\right) d\mathbf{x} \\ &= \int_R \exp\left(-\frac{1}{2k_\mathrm{B}T} \sum_{i=1}^{2N} \lambda_i^{\mathrm{M}} q_i^2\right) d\mathbf{q} \, e^{-E^{\mathrm{M}}/k_\mathrm{B}T} \\ &= V^{\mathrm{M}} \frac{\sqrt{2k_\mathrm{B}T}^{\,P^{\mathrm{M}}}}{\prod_i' \sqrt{\lambda_i^{\mathrm{M}}}} \int_R \exp\left(-\sum_i{}' x_i^2\right) d\mathbf{x} \, e^{-E^{\mathrm{M}}/k_\mathrm{B}T} \\ &= V^{\mathrm{M}} \frac{\sqrt{2\pi k_\mathrm{B}T}^{\,P^{\mathrm{M}}}}{\prod_i' \sqrt{\lambda_i^{\mathrm{M}}}} \, e^{-E^{\mathrm{M}}/k_\mathrm{B}T} \,, \end{aligned} \tag{1.39}$$

where the primes again denote positive eigenvalues, in this case indexing over $[1 + N_0^{\mathrm{M}}, 2N]$, and we again pulled out the zero mode contributions, which are in this case

$$V^{\mathrm{M}} = \int\limits_{0}^{\infty} \prod_{i=1}^{N_0^{\mathrm{M}}} dq_i \,. \tag{1.40}$$

With $\Delta E = E^{\mathrm{SP}} - E^{\mathrm{M}}$ we therefore get

$$\Gamma^{\mathrm{HTST}} = \frac{1}{2\pi} \sqrt{2\pi k_\mathrm{B}T}^{\,N_0^{\mathrm{M}} - N_0^{\mathrm{SP}}} \frac{V^{\mathrm{SP}}}{V^{\mathrm{M}}} \frac{\prod_i' \sqrt{\lambda_i^{\mathrm{M}}}}{\prod_i' \sqrt{\lambda_i^{\mathrm{SP}}}} \sqrt{\sum_i{}' \frac{a_i^2}{\lambda_i^{\mathrm{SP}}}} \, e^{-\Delta E/k_\mathrm{B}T} \,. \tag{1.41}$$

This finally brings us back to the transition rate (1.21), as we can compactly write

$$\Gamma^{\mathrm{HTST}} = \Gamma_0 e^{-\Delta E/k_\mathrm{B}T} \,, \quad \Gamma_0 = \frac{1}{2\pi} \frac{V^{\mathrm{SP}}}{V^{\mathrm{M}}} \nu \Omega_0 \sqrt{2\pi k_\mathrm{B}T}^{\,N_0^{\mathrm{M}} - N_0^{\mathrm{SP}}} \,, \tag{1.42}$$

where Γ_0 is the attempt rate and we factored out the dynamical factor

$$v = \sqrt{\sum_i{}' \frac{a_i^2}{\lambda_i^{\mathrm{SP}}}} \tag{1.43}$$

and the curvature factor

$$\Omega_0 = \sqrt{\frac{\det' H^{\mathrm{M}}}{\det' H^{\mathrm{S}}}} = \sqrt{\frac{\prod_i' \lambda_i^{\mathrm{M}}}{\prod_i' \lambda_i^{\mathrm{SP}}}}\,. \tag{1.44}$$

Note a subtle difference of in the definition of Ω_0 between Langers theory (see Ref. [144]) and HTST, where the former also includes the absolute value of the negative curvature at the saddle point.

1.3.3 *Entropy contributions*

Refs. [40, 144] (see also [156]) draw a relation between the attempt rate Γ_0 and the entropy. It can be shown that transition rates of individual processes can be dominantly determined by a part of the prefactor, which can be interpreted as a measure of entropy.

The partition function of a classical system of N spins in a canonical ensemble reads

$$Z = (4\pi)^{-N} \int_{\Omega_1} \cdots \int_{\Omega_N} \prod_i d\Omega_i e^{-E/k_{\mathrm{B}}T}\,, \tag{1.45}$$

where $d\Omega_i$ is the element of solid angle in the direction of the i^{th} spin. By applying the same tools as in the previous derivations, the partition function, expanded around a stationary point of the energy landscape (see Equation (1.25)), can be expressed as

$$\begin{aligned} Z &= \frac{e^{-E/k_{\mathrm{B}}T}}{(4\pi)^N} \int \prod_i dq_i e^{-\frac{1}{2}\sum \lambda_i q_i^2/k_{\mathrm{B}}T} \\ &= \frac{e^{-E/k_{\mathrm{B}}T}}{(4\pi)^N} \prod_i \int dq_i e^{-\frac{1}{2}\lambda_i q_i^2/k_{\mathrm{B}}T}\,. \end{aligned} \tag{1.46}$$

The integration over an unstable mode direction, one of which will appear for the saddle point configuration, can be simplified by assuming constant energy for the width σ of the infinitessimally thin transition state, i.e. we set $\lambda_1 \to 0$ in that case. And with the same considerations of integration over zero modes as before, one obtains

$$Z = \frac{\sigma V e^{-E/k_{\mathrm{B}}T}}{(4\pi)^N} \prod_i{}' \sqrt{\frac{2\pi k_{\mathrm{B}}T}{\lambda_i}}\,, \tag{1.47}$$

where V is the zero mode volume, σ the unstable mode volume and $\prod'$ again denotes the product of only positive eigenvalues. Note that analogous to before, $V = 1$ if there are no zero modes and $\sigma = 1$ if there are no unstable modes. At the saddle point configuration there should be exactly one unstable mode of negative curvature $\lambda_1 < 0$, while there should be none at the minimum. The Helmholtz free energy of each state can be estimated as

$$\begin{aligned} F &= -k_\mathrm{B} T \ln(Z) \\ &= E - k_\mathrm{B} T \ln(\sigma V) + N k_\mathrm{B} T \ln(4\pi) - k_\mathrm{B} T \sum_i{}' \ln\left(\sqrt{\frac{2\pi k_\mathrm{B} T}{\lambda_i}}\right). \end{aligned} \tag{1.48}$$

The entropy, in turn, can then be derived as (compare Eqs. (I.30) – (I.32))

$$S = -\frac{\partial F}{\partial T} = k_\mathrm{B} \ln(\sigma V) - N k_\mathrm{B} \ln(4\pi) + k_\mathrm{B} \sum_i{}' \ln\left(\sqrt{\frac{2\pi k_\mathrm{B} T}{\lambda_i}}\right) + k_\mathrm{B} \frac{P}{2}, \tag{1.49}$$

where P is the number of positive eigenvalues. This gives an entropy difference between the transition state and the (meta)stable state of

$$\frac{\Delta S}{k_\mathrm{B}} = \frac{N_0^\mathrm{M} - N_0^\mathrm{SP} - 1}{2} + \ln \frac{\sigma^\mathrm{SP} V^\mathrm{SP}}{V^\mathrm{M}} + \ln\left(\sqrt{2\pi k_\mathrm{B} T}^{N_0^\mathrm{M} - N_0^\mathrm{SP} - 1} \sqrt{\frac{\prod_i' \lambda_i^\mathrm{M}}{\prod_i' \lambda_i^\mathrm{SP}}} \right), \tag{1.50}$$

where we used the fact that $\sigma^\mathrm{M} = 1$, as there are no negative curvatures at the local minimum. Therefore

$$e^{\Delta S/k_\mathrm{B}} = e^{(N_0^\mathrm{M} - N_0^\mathrm{SP} - 1)/2} \frac{\sigma^\mathrm{SP} V^\mathrm{SP}}{V^\mathrm{M}} \sqrt{2\pi k_\mathrm{B} T}^{N_0^\mathrm{M} - N_0^\mathrm{SP} - 1} \sqrt{\frac{\prod_i' \lambda_i^\mathrm{M}}{\prod_i' \lambda_i^\mathrm{SP}}}, \tag{1.51}$$

by which we can relate the attempt rate Γ_0 to the configurational entropy as

$$\Gamma_0 = \frac{e^{-(N_0^\mathrm{M} - N_0^\mathrm{SP} - 1)/2}}{\sigma^\mathrm{SP}} \nu \sqrt{2\pi k_\mathrm{B} T} e^{\Delta S/k_\mathrm{B}}, \tag{1.52}$$

meaning there is a direct relation between the rate prefactor and the difference in entropy between the local minimum and the saddle point.

1.3.4 *Zero mode contributions*

These derivations are found in the supplementary material to Ref. [38]. For zero modes (meaning modes with zero eigenvalue), often also called Goldstone modes, we get contributions to the rate prefactor Γ_0 of Equation (1.29) in form of integrals over such modes,

$$V_q = \int dq\,. \tag{1.53}$$

Integration of the distribution function over a Goldstone mode is non-Gaussian and results in the volume of the mode. For a skyrmion in a magnetic strip, the Goldstone modes correspond to pure translations of the magnetic configuration. Consequently, the volumes of the modes can be obtained by integration over a spatial coordinate. Localized magnetic textures, such as the skyrmion or its saddle point state for the radial collapse, can be described by a set of unit vectors $M = \{\vec{n}_1, \vec{n}_2, \ldots, \vec{n}_N\}$, where N is the number of spins in the system. Let $\vec{r}$ be the position of the localized state, i.e. the position of the skyrmion center. Translation of the magnetic texture along a direction $\vec{e}_a$ by an infinitessimal distance da can be described by

$$M(\vec{r} + \vec{e}_a da) - M(\vec{r}) = \frac{dM}{da} da\,. \tag{1.54}$$

This is equivalent to the displacement along a zero mode Λ_i, meaning we can also write

$$M(\vec{r} + \vec{e}_a da) - M(\vec{r}) = \Lambda_i(\vec{r}) dq\,. \tag{1.55}$$

These relations are schematically visualised in Figure 1.23.

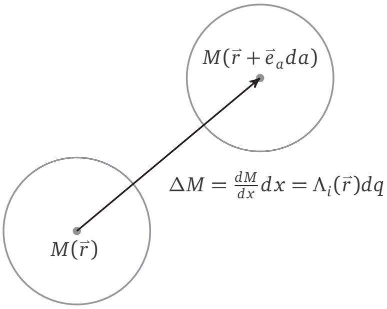

$M(\vec{r} + \vec{e}_a da)$
$\Delta M = \frac{dM}{dx} dx = \Lambda_i(\vec{r}) dq$
$M(\vec{r})$

FIGURE 1.23 – Schematic visualisation of translational skyrmion mode

Schematic visualisation of the displacement of a skyrmion spin configuration. The skyrmion core is shifted by a finite translation $\vec{e}_a da$.

As, by our definition, the zero mode is normalized, i.e. $|\Lambda_i| = 1$, we can reformulate this in absolute values and write

$$dq = |M(\vec{r} + \vec{e}_x dx) - M(\vec{r})| = \left| \frac{dM(\vec{r})}{dx} \right| dx\,. \tag{1.56}$$

Specifically, if $\vec{e}_a$ is chosen to be along the lattice vector $\vec{a}$

$$\left| \frac{dM(\vec{r})}{dx} \right| \approx \frac{1}{a} |M(\vec{r} + \vec{a}) - M(\vec{r})| = \frac{1}{a} \left(\sum_i^N |\vec{n}_i(\vec{r} + \vec{a}) - \vec{n}_i(\vec{r})|^2 \right)^{1/2}, \tag{1.57}$$

where a is the lattice constant. We can therefore replace the integration over the eigenmode by an integration over the spatial coordinate

$$V_q = \int dq = \int_0^{L_a} \left| \frac{dM(\vec{r})}{dx} \right| dx = \frac{L_a}{a} \left(\sum_i^N |\vec{n}_i(\vec{r}+\vec{a}) - \vec{n}_i(\vec{r})|^2 \right)^{1/2} . \tag{1.58}$$

Note that special considerations need to be made for these calculations in order to obtain physical results. For example, the system size L_a may be infinite, for example under periodical boundary conditions, meaning that the fact that there are also infinitely many skyrmion collapse saddle points needs to be taken into account in order to obtain a finite lifetime. Another example is the fact that, strictly speaking, a (numerically) zero mode corresponding to a translation of a skyrmion in a non-periodical direction is not a zero mode.

1.4 THE DYNAMICAL MATRIX

The dynamical prefactor v (see Eqs. (1.42) and (1.43)) takes into account the dynamics of the system at the saddle point and is determined in part by the set of N deterministic LLG equations, linearized around the saddle point (see Equation (1.28)), which read in index notation

$$\partial_t n^i_\alpha = \mathcal{V}^{ij}_{\alpha\beta} n^j_\beta \, . \tag{1.59}$$

To ease notation, we split the dynamical Equation (1.28) into precession and damping contributions

$$\mathcal{V}|^{ij} = \frac{\gamma}{(1+\alpha^2)\mu^i_{\mathrm{S}}} \mathcal{V}^{\mathrm{P}}|^{ij} + \frac{\gamma\alpha}{(1+\alpha^2)\mu^i_{\mathrm{S}}} \mathcal{V}^{\mathrm{D}}|^{ij} \, . \tag{1.60}$$

Starting out with the precession contribution:

$$\mathcal{V}^{\mathrm{P}}|^{ij}_{\alpha\beta} n^j_\beta = -\varepsilon_{\alpha\mu\nu} n^i_\mu \frac{\partial E}{\partial n^i_\nu} \, , \tag{1.61}$$

we can write within the harmonic approximation (1.25), where $\partial G/\partial n = 0$ and $\partial H/\partial n = 0$,

$$\begin{aligned} \frac{\partial E}{\partial n^i_\nu} &= \frac{\partial}{\partial n^i_\nu} \left(E_0 + G^k_\alpha n^k_\alpha + \frac{1}{2} n^k_\alpha H^{kl}_{\alpha\beta} n^l_\beta \right) \\ &= G^i_\nu + \frac{1}{2}(H^{il}_{\nu\beta} n^l_\beta + n^k_\alpha H^{ki}_{\alpha\nu}) \\ &= G^i_\nu + H^{ik}_{\nu\gamma} n^k_\gamma \, . \end{aligned} \tag{1.62}$$

Note that $G = 0$ at stationary points and can therefore be omitted for HTST calculations. Using Equation (1.62), we can rewrite Equation (1.61) as

$$\mathcal{V}^{\mathrm{P}}|^{ij}_{\alpha\beta} n^j_\beta = \varepsilon_{\alpha\mu\nu} n^i_\mu \left(G^i_\nu + H^{ik}_{\nu\gamma} n^k_\gamma \right) . \tag{1.63}$$

Furthermore, the linearised LLG equation implies $\partial \mathcal{V} / \partial n = 0$ and therefore, taking the derivative of Equation (1.63), we can write

$$\mathcal{V}^{\mathrm{P}}|^{ij}_{\alpha\beta} = \frac{\partial}{\partial n^j_\beta} \varepsilon_{\alpha\mu\nu} n^i_\mu \left(G^i_\nu + H^{ik}_{\nu\gamma} n^k_\gamma \right) . \tag{1.64}$$

It finally follows that the precession contribution to the velocity matrix has the form

$$\mathcal{V}^{\mathrm{P}}|^{ij}_{\alpha\beta} = \varepsilon_{\alpha\mu\nu} n^i_\mu H^{ij}_{\nu\beta} + \delta_{ij} \varepsilon_{\alpha\beta\nu} \left(G^i_\nu + H^{ik}_{\nu\gamma} n^k_\gamma \right) . \tag{1.65}$$

For the damping contribution we follow the same procedure, except that we can make use of $\vec{a} \times (\vec{b} \times \vec{c}) = \vec{b}(\vec{c} \cdot \vec{a}) - \vec{c}(\vec{a} \cdot \vec{b})$ to rewrite the triple product

$$\vec{n}^i \times \left(\vec{n}^i \times \frac{\partial E}{\partial \vec{n}^i} \right) = \vec{n}^i \left(\vec{n}^i \cdot \frac{\partial E}{\partial \vec{n}^i} \right) - \frac{\partial E}{\partial \vec{n}^i} \tag{1.66}$$

and thereby obtain

$$\begin{aligned} \mathcal{V}^{\mathrm{D}}|^{ij}_{\alpha\beta} n^j_\beta &= - \varepsilon_{\alpha\mu\nu} n^i_\mu \varepsilon_{\nu\gamma\rho} n^i_\gamma \frac{\partial E}{\partial n^i_\rho} \\ &= \frac{\partial E}{\partial n^i_\alpha} - n^i_\alpha n^i_\nu \frac{\partial E}{\partial n^i_\nu} \\ \Rightarrow \quad \mathcal{V}^{\mathrm{D}}|^{ij}_{\alpha\beta} &= \frac{\partial}{\partial n^j_\beta} \left(G^i_\alpha + H^{ik}_{\alpha\gamma} n^k_\gamma - n^i_\alpha n^i_\nu \left(G^i_\nu + H^{ik}_{\nu\gamma} n^k_\gamma \right) \right) \\ &= H^{ij}_{\alpha\beta} - \delta_{ij} \delta_{\alpha\beta} n^i_\nu \left(G^i_\nu + H^{ik}_{\nu\gamma} n^k_\gamma \right) - \delta_{ij} n^i_\alpha \left(G^i_\beta + H^{ik}_{\beta\gamma} n^k_\gamma \right) \\ &\quad - n^i_\alpha n^i_\nu H^{ij}_{\nu\beta} . \end{aligned} \tag{1.67}$$

In the frame of HTST, the Hessian matrix – needed as a contribution to the dynamical matrix – only needs to be calculated at extremal points. To take the curvature of the spin manifold into account, in this case, a derivation from Lagrange multipliers, as shown in Appendix C.4, is sufficient. It gives a simple correction on the diagonal

$$H = \bar{H} - \mathbb{1} \left(x_j \cdot \frac{\partial \mathcal{H}}{\partial \vec{n}_j} \right) . \tag{1.68}$$

A generalized derivation of the Hessian matrix is shown in the following Section 2.2, where the Hessian is needed also in non-stationary points.

As the Hessian (1.68) is given in the embedding space $\mathcal{E}$, it includes unphysical degrees of freedom which are parallel to the spins (see also Section I.2). In order to avoid these non-tangential degrees of freedom, we again transform into the tangent space basis T (see also Equation (I.10))

$$\mathcal{V}_{ij} \rightarrow T_i^T \mathcal{V}_{ij} T_j \,. \tag{1.69}$$

Note Equation (1.31), giving the representation of the dynamical matrix in the tangent space:

$$\mathcal{V}|^{2N} = \Lambda^T|^{2N} T^T \mathcal{V}|^{3N} T \Lambda|^{2N} \,. \tag{1.70}$$

Of course, one can alternatively use the $3N$ representation of the Hessians eigenbasis:

$$\mathcal{V}|^{2N} = \Lambda^T \mathcal{V} \Lambda \,. \tag{1.71}$$

1.5 EIGENMODES OF THE DYNAMICAL MATRIX

Note that the basis for this section was already covered, for example by David Bauer [53] and transferred to non-collinear spin configurations, for example by Levente Rózsa [119] (see ref. [157] for a more recent study of dynamical eigenmodes of skyrmions). However, the following will be written in the external frame of reference (see also Section 2.2). In the author's opinion, this approach gives a far more natural solution and intuitive understanding, as well as simpler equations. Furthermore, it is not necessary to only consider stationary points for the calculation of these quantities.

We are interested in the dynamical eigenmodes $\vec{n}_i(t)$ of a spin system. These can be determined from the linearised LLG Equation (1.28), which, can be conveniently rewritten

$$\partial_t \vec{n}_i = \mathcal{V}_{ij} \vec{n}_j = \mathcal{E}_{ik} D_{kl} \mathcal{E}_{lj}^{-1} \vec{n}_j \,, \tag{1.72}$$

where D is a matrix with the eigenvalues ϵ_i on its diagonal and $\mathcal{E}$ is the tangential part of the eigenbasis of the dynamical matrix $\mathcal{V}$ and is taken to be a $3N \times 2N$ matrix, i.e. consisting of the $2N$ eigenvectors tangential to the spin configuration. The eigenbasis $\mathcal{E}$ is known from the diagonalization of the $2N$ representation $\mathcal{V}|^{2N}$ and subsequent transformation into the embedding space $\mathcal{E}|^{3N} = T \mathcal{E}|^{2N} T^T$.

If we use the notation of small tangential perturbations $n(t) = n(t_0) + q(t)$, analogous to the expansion performed in Section 1.3 (see Equations (1.26) and (1.27)), we know that $n(t_0)$ is by construction not in the tangent space, i.e. $\mathcal{V} n(t_0) = 0$ and $\mathcal{E} n(t_0) = 0$ and we can therefore write

$$\begin{aligned} &\partial_t \mathcal{E}_{ij}^{-1} \vec{q}_j = D_{ik} \mathcal{E}_{kj}^{-1} \vec{q}_j = \epsilon_i \mathcal{E}_{ij}^{-1} \vec{q}_j \\ &\Rightarrow \mathcal{E}_{ij}^{-1} \vec{q}_j = c_i e^{\epsilon_i t} \\ &\Leftrightarrow \vec{q}_i(t) = c_i \mathcal{E}_i e^{\epsilon_i t} \,, \end{aligned} \tag{1.73}$$

where $\mathcal{E}_i$ denotes one eigenvector, i.e. a column of the eigenbasis matrix $\mathcal{E}$.

This means that the general solution is

$$n(t) = n(t_0) + q(t), \quad q_i(t) = c_i \mathcal{E}_i e^{\epsilon_i t}\,, \tag{1.74}$$

where $\epsilon_i = \gamma_i + i\omega_i$ are the eigenvalues.

However, we can simplify the general solution to the purely real eigenmodes. As $\mathcal{V}$ is in general a real matrix with complex eigenvalues and eigenvectors, we can write for any eigenvalue and -vector pair $\{\epsilon_i, \mathcal{E}_i\}$

$$\mathcal{V}\mathcal{E}_i = \epsilon_i \mathcal{E}_i \Rightarrow (\mathcal{V}\mathcal{E}_i)^* = (\epsilon_i \mathcal{E}_i)^* \iff \mathcal{V}\mathcal{E}_i^* = \epsilon_i^* \mathcal{E}_i^*\,. \tag{1.75}$$

This means that the conjugate pair $\{\epsilon_j = \epsilon_i^*, \mathcal{E}_j = \mathcal{E}_i^*;\ i \neq j\}$ is also an eigenmode. Due to the structure of the differential Equation (1.73) for the perturbations, all linear combinations of eigenvectors are still eigenvectors. We therefore choose q to be the real solution

$$q(t) = \frac{1}{2}\left(\mathcal{E}_i e^{\epsilon t} + \mathcal{E}_i^* e^{\epsilon^* t}\right) = e^{\gamma t}\left(\cos(\omega t)\,\Re(\mathcal{E}_i) + \sin(\omega t)\,\Im(\mathcal{E}_i)\right)\,. \tag{1.76}$$

Note that due to $\epsilon_i = \epsilon_j^*$, it is sufficient to calculate only half the eigenmodes, namely those with the highest or lowest imaginary part, as the counterpart has a flipped sign.

While the above may seem somewhat abstract, the resulting eigenmodes are very intuitive to understand when visualised. The case of a damped single spin at a local minimum is shown in Figure 1.24, where – starting from an initial displacement – the spin performs a damped, elliptic eigenprecession back to the minimum. In the case of a saddle point as starting point, there may be an exponentially divergent contribution from the negative real part γ_i of the eigenvalue ϵ_i.

1.5.1 *Simplification in the precession-only case*

When the damping term in the dynamical matrix is neglected, one of course gets only stable eigenmodes, which do not decay or diverge. This can be shown by the fact that $\mathcal{V}$ is in this case skew-symmetric. We can therefore write

$$\mathcal{V}n = \epsilon n \Rightarrow n^\dagger \mathcal{V} n = \epsilon |n|^2\,, \tag{1.77}$$

where † denotes the Hermitian conjugate (i.e. complex conjugation and transposition) and $|n|$ is of course a real number. Thus,

$$\epsilon^* |n|^2 = (n^\dagger \mathcal{V} n)^\dagger = n^\dagger \mathcal{V} (n^\dagger)^\dagger = -n^\dagger \mathcal{V} n = -\epsilon |n|^2\,. \tag{1.78}$$

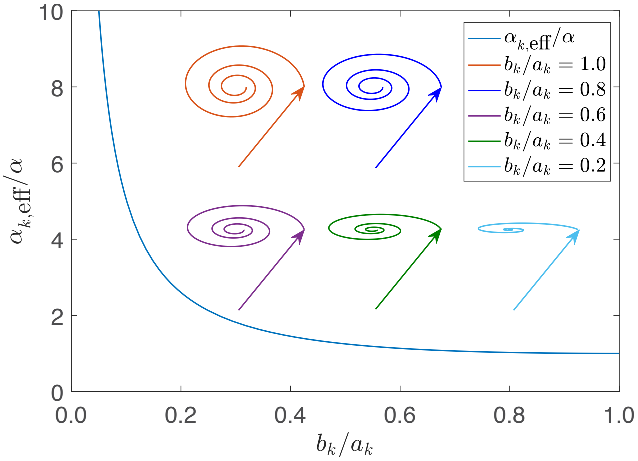


FIGURE 1.24 – Dynamical eigenmode trajectories of a single spin

Effective damping parameter α_k^{eff} as a function of the inverse aspect ratio b_k/a_k of the polarisation ellipse of the eigenmode. The insets illustrate the corresponding damped precession of a spin, corresponding to what is described in this chapter for the case of finite damping at a local minimum. Reprinted with permission from [158]. Copyright 2018 American Physical Society.

This means that $\epsilon^* = -\epsilon$ and therefore $\mathcal{V}$ has only imaginary eigenvalues, i.e. $\epsilon_i = i\omega_i$. The dynamical eigenmodes (1.76) hence simplify to stable solutions

$$q(t) = \cos(\omega t)\,\Re(\mathcal{E}_i) + \sin(\omega t)\,\Im(\mathcal{E}_i)\,. \tag{1.79}$$

The case without damping is shown in Figure 1.25 for the more complex system containing an isolated skyrmion. Though the modes are no longer trivial to plot, there is a set of eigenfrequencies which distincly belong to the skyrmion (not the ferromagnetic background state), corresponding to symmetry-related transformations of the skyrmion: translation, core gyration, and elliptic and higher-order deformations.

The skyrmion is shown to have a breathing mode, a translational mode and an elliptical, a triagonal, a quadruple and higher-order eigenmodes. There is a distinct similarity to the eigenmodes of the Hessian, but from the derivations in this chapter it is not clear if their relation can be quantitatively shown.

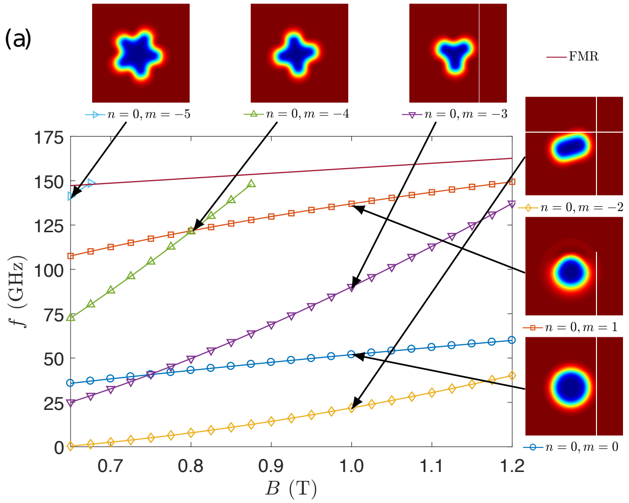


FIGURE 1.25 – Dynamical skyrmion modes

Frequencies of localized spin wave modes at zero damping $\alpha = 0$. The six lowest eigenmodes of a skyrmion are visualised by colour-coding the out of plane spin component of the configuration displaced along the mode. Reprinted with permission from [157]. Copyright 2018 American Physical Society.

1.6 TRANSITION RATE TEST CASE: 2D SKYRMIONS

Due to the combination of different methods in *Spirit* [47, 48], it is very easy to calculate the rates of transitions:

- calculate minima using direct minimisation
- calculate transition path and saddle point using GNEB
- calculate the transition rate prefactor using HTST

To extract the prefactor, one simply needs a chain containing the local minimum and saddle point images:

```
NOI=10
with state.State(input_file, quiet=True) as p_state:
    configuration.plus_z(p_state)
    chain.image_to_clipboard(p_state)
    chain.set_length(p_state, NOI)

    configuration.skyrmion(p_state, 5.0, phase=-90.0, idx_image=0)
    configuration.plus_z(p_state, idx_image=NOI-1)

    simulation.start(p_state, LLG, VP, idx_image=0)

    transition.homogeneous(p_state, 0, NOI−1)
    chain.update_data(p_state)
    parameters.gneb.set_image_type_automatically(p_state)
    simulation.start(p_state, GNEB, VP)

    energies = chain.get_energy(p_state)
    idx_max = energies.index(max(energies))

    # returns value in 1/(Ks)
    prefactor = quantities.get_htst_prefactor(p_state, idx_image_minimum, idx_image_sp)
```

LISTING 1.2 –

Combination of LLG, GNEB and HTST used to get the transition rate prefactor

The calculations in this thesis have been verified against the *UppASD* and *Matjes* codes.

1.6.1 *Simple cubic lattice*

In order to test the implementation in *Spirit*, the following example is presented. Parameters are chosen to be

$$\mu_S = 3\mu_B\,, \quad B^{\mathrm{ext}} = 0.2\ \mathrm{T}\,, \quad K = 0.5\ \mathrm{T}\,, \quad J_{ij} = 20\ \mathrm{meV}\,, \quad D_{ij} = 2\ \mathrm{meV}\,. \qquad (1.80)$$

The lattice is simple cubic and of size 50 × 50 with periodical boundaries.

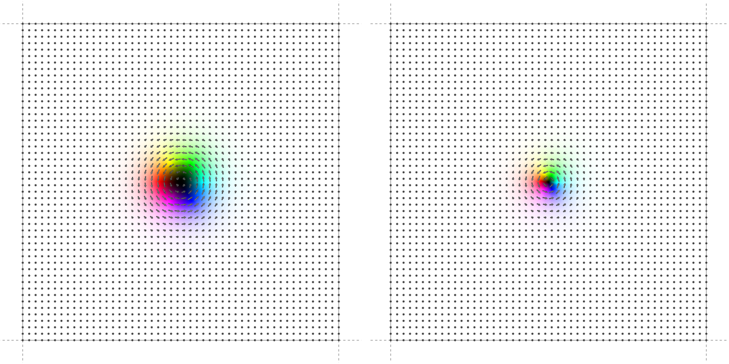


FIGURE 1.26 – Skyrmion and collapse saddle point in simple cubic lattice

Simple cubic lattice with periodical boundary conditions, containing a single Bloch skyrmion (left) and the saddle point configuration for the radial skyrmion collapse (right).

minimum	saddle point
$1.652613 \cdot 10^{-09}$	$-3.682553 \cdot 10^{-01}$
$1.652870 \cdot 10^{-09}$	$1.732453 \cdot 10^{-03}$
$1.137551 \cdot 10^{-01}$	$1.732453 \cdot 10^{-03}$
$9.639774 \cdot 10^{-01}$	$1.026862 \cdot 10^{+00}$
$1.023241 \cdot 10^{+00}$	$1.026862 \cdot 10^{+00}$
$1.023241 \cdot 10^{+00}$	$1.061867 \cdot 10^{+00}$
$1.328575 \cdot 10^{+00}$	$1.324041 \cdot 10^{+00}$
$1.328575 \cdot 10^{+00}$	$1.324041 \cdot 10^{+00}$
$1.354243 \cdot 10^{+00}$	$1.357010 \cdot 10^{+00}$
$1.361490 \cdot 10^{+00}$	$1.357010 \cdot 10^{+00}$

TABLE 1.2 – Lowest eigenvalues of the skyrmion collapse (sc lattice)

As expected, the saddle point has one negative eigenvalue. The minimum has two zero modes for the translation of the skyrmion, whereas those two modes have grown in eigenvalue at the saddle point, due to the stronger curl in the magnetisation, relative to the lattice. Values are given in meV.

We recall Equation (1.42):

$$\Gamma^{\mathrm{HTST}} = \frac{\nu}{2\pi}\Omega_0 e^{-\Delta E/k_{\mathrm{B}}T} \,,$$

where the dynamical prefactor ν is given by Equation (1.43)

$$\nu = \sqrt{2\pi k_{\mathrm{B}} T}^{P^{\mathrm{SP}}-P^{\mathrm{M}}} \frac{V^{\mathrm{SP}}}{V^{\mathrm{M}}} \underbrace{\sqrt{\sum_i{}' \frac{a_i^2}{\lambda_i^{\mathrm{SP}}}}}_{s} \,.$$

s	ν	c_x	c_y	c_z
$3.9076 \cdot 10^{-1}$	$1.3747 \cdot 10^{1}$	$1 \cdot 10^{0}$	$1.5391 \cdot 10^{-2}$	$1.2158 \cdot 10^{2}$

TABLE 1.3 – Prefactor contributions for the skyrmion collapse (sc lattice)

As expected, the saddle point has one negative eigenvalue. The minimum has two zero modes for the translation of the skyrmion, whereas those two modes have grown in eigenvalue at the saddle point, due to the stronger curl in the magnetisation and relative to the lattice.

At the saddle point, one eigenvalue has turned negative, but due to the discretisation the zero modes are no longer present. The prefactor is then $\Gamma_0 = 9.05966845952 \cdot 10^{11} \, \frac{1}{\mathrm{K \cdot s}}$.

1.6.2 *Hexagonal lattice*

Hexagonal lattice (here chosen as 120 deg). Size of the system: 60x60 translations along the lattice vectors, periodical boundaries.

$$\mu_S = 3\mu_B\,, \quad B^{\text{ext}} = 2\text{T}\,, \quad K = 0.4\ \text{meV}\,, \quad J_{ij} = 5.9\ \text{meV}\,, \quad D_{ij} = 1.6\ \text{meV}\,. \tag{1.81}$$

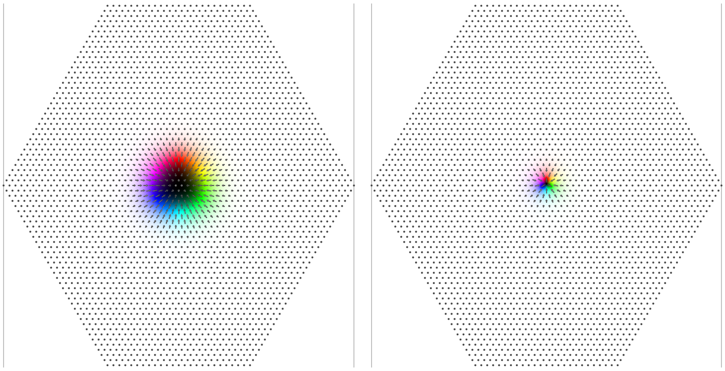


FIGURE 1.27 – Skyrmion and collapse saddle point in hexagonal lattice

Hexagonal lattice (120°) with periodical boundary conditions, containing a single Neel skyrmion (left) and the saddle point configuration for the radial skyrmion collapse (right).

minimum	saddle point
$-5.660782 \cdot 10^{-09}$	$-5.116295 \cdot 10^{+00}$
$8.525432 \cdot 10^{-09}$	$2.124529 \cdot 10^{-01}$
$1.948042 \cdot 10^{-01}$	$2.124529 \cdot 10^{-01}$
$6.586385 \cdot 10^{-01}$	$1.148786 \cdot 10^{+00}$
$6.586385 \cdot 10^{-01}$	$1.148786 \cdot 10^{+00}$
$1.158167 \cdot 10^{+00}$	$1.258589 \cdot 10^{+00}$
$1.158167 \cdot 10^{+00}$	$1.270695 \cdot 10^{+00}$
$1.273047 \cdot 10^{+00}$	$1.270695 \cdot 10^{+00}$
$1.273047 \cdot 10^{+00}$	$1.273199 \cdot 10^{+00}$
$1.277477 \cdot 10^{+00}$	$1.276659 \cdot 10^{+00}$

TABLE 1.4 – Lowest eigenvalues of the skyrmion collapse (hexagonal lattice)

Expectedly, the qualitative behaviour matches Table 1.2. The saddle point has one negative eigenvalue, while the minimum has two translational zero modes. At the saddle point, those zero modes vanish due to the stronger curl in the magnetisation, relative to the lattice. Values are given in meV.

The prefactor, which is extracted from *Spirit* needs to be scaled by the number of occurences per unit cell of the corresponding saddle point, which in this case is 2. This is schematically shown in the following Figure 1.28, illustrating the possible positions of the centre of the saddle point skyrmion configuration.

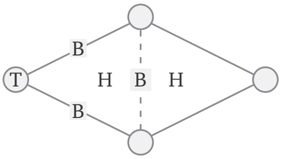


FIGURE 1.28 – Schematic of skyrmion collapse saddle points in the hexagonal unit cell

Schematic visualisation of the various skyrmion collapse saddle points, which can be calculated using the GNEB method. At the given field of $B = 0.2$ T, the "hollow" (H) configuration (see Figure 1.27) is a first order saddle point, while "bridge" (B) and "top" (T) are not.

Therefore, the prefactor is $\Gamma_0 = 2 \cdot 4.523811 \cdot 10^{10} \frac{1}{\mathrm{K \cdot s}} = 9.047622 \cdot 10^{10} \frac{1}{\mathrm{K \cdot s}}$.

1.6.3 *Arrhenius plot of the radial collapse*

Using the temperature-dependent prefactor and the barrier- and temperature-dependent exponential, one can create an Arrhenius plot of the lifetimes of a skyrmion in a periodical system over the temperature and one parameter. The choice of parameters for this example is $J = 1$ and $D = 0.6$, while the magnetic field is varied between 3.5 T and 5 T. This case is easily calculated, as there is only one relevant transition for the skyrmion: its radial collapse. As Figure 1.29 shows, the temperature- and field-dependence is quite simple. Note that there are two translational zero modes at the minimum, while – due to the lattice discretisation and the defect-like shape of the skyrmion at the saddle point – there are no zero modes at the saddle point. Consequently, the transition rate prefactor has a linear temperature dependence.

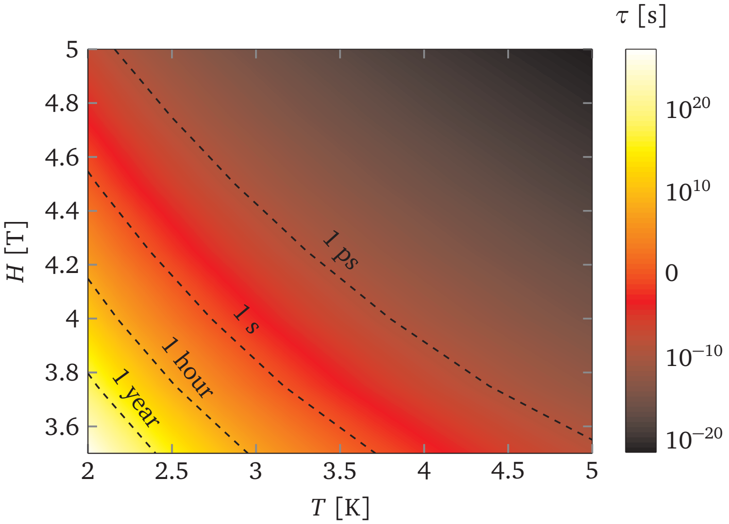


FIGURE 1.29 – HTST Arrhenius plot for the skyrmion collapse

The lifetime τ of an isolated skyrmion in a periodical two-dimensional system in dependence on temperature T and external magnetic field H. Only the collapse is considered as a possible transition, so the plot is quite simple. Isolines for lifetimes between 1 ps and 1 year are shown. For the given parameters, the lifetime varies by 20 orders of magnitude within a range of only 2 K. This figure has been published in [47].

1.7 RATE THEORY APPLIED TO A SKYRMION ON A RACETRACK

This theory can consequently be applied to practical problems, such as the design of skyrmion racetrack devices. Here, the calculation of skyrmion lifetimes for a racetrack configuration is demonstrated, making use of the multiscale approach mentioned in Section I.2 by taking atomistic parameters from first principles. Note that, while the author collaborated on Ref. [38], the following results were obtained by P. Bessarab using UppASD. The paper presents the predicted lifetime of a skyrmion in a periodically continued racetrack as a function of temperature, magnetic field and track width and shows consistency between prediction and experimental data. The model system chosen is a Pd/Fe bilayer on an Ir(111) substrate, a well-studied system for which accurate parameters for the Heisenberg model are known from ab initio calculations.

As the racetrack has a finite width, the skyrmion can escape through the boundary of the system. Due to the skyrmion hall effect [24, 159], which pushes the skyrmions towards the system boundary, potentially annihilating it [21, 160], this is an important transition to consider for the skyrmion lifetime. In the paper [38] it is shown that the escape through the track boundary dominates at low external magnetic field, but above a certain crossover field, the collapse inside the track becomes more relevant. This crossover stems from the contribution of the pre-exponential factor in the Arrhenius law (1.21), which shows different dependencies on the external magnetic field for the escape transition and for the collapse transition. These results are shown in Figure 1.30.

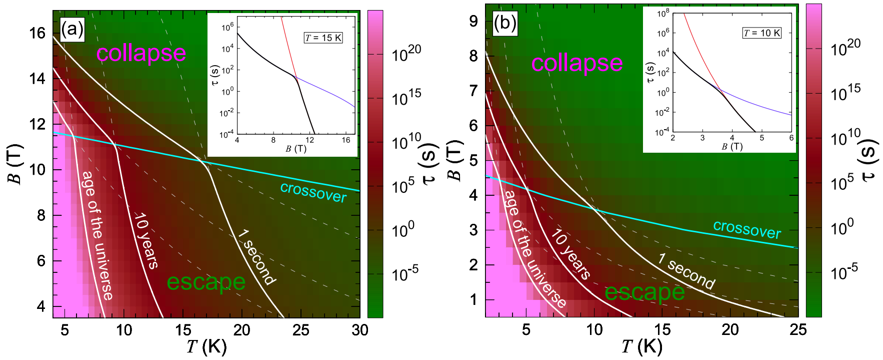


FIGURE 1.30 – Arrhenius plot for the lifetime of racetrack skyrmions

Lifetime of a skyrmion in a Pd/Fe/Ir(111) racetrack. Contour plot of the calculated lifetime of an isolated skyrmion in a 23.5nm wide strip as a function of applied magnetic field strength and temperature, shown for the fcc (a) and hcp (b) stacking of the Pd layer. White contour lines have a characteristic cusp due to the crossover between collapse and escape mechanism indicated by the cyan line. Above the crossover line, the skyrmion lifetime is mostly defined by the collapse mechanism, but the escape mechanism dominates below the crossover line. White dashed lines are contour lines of the collapse and escape lifetimes. Insets show the cut of the contour plot at $T = 15$ K (a) and $T = 10$ K (b). In the insets, the annihilation time due to collapse in the interior and escape through the boundary, as well as the total skyrmion lifetime are shown as red, purple and black curves respectively. The data was gathered and plotted by Pavel Bessarab. Reprinted from [38] with minor modifications, licensed under CC BY 4.0.

The crossover between the two annihilation mechanisms can be explained mostly with the crossover of the energy barriers, which in turn can be interpreted as an effect of the skyrmion radius r_S. The energy barrier of the radial collapse is directly proportional to the number of spins, which need to be flipped, and therefore scales with the skyrmions area $\propto\sim r_{\mathrm{S}}^2$. Consequently, as the skyrmion configuration shrinks proportionally with higher magnetic fields, the barrier tends to zero for large fields, where the equilibrium skyrmion approaches the size of the saddle point configuration, which is almost independent of the field. The escape barrier, however, is defined by the number of spins at the boundary, which need to be untwisted. It therefore scales with the skyrmions diameter $\propto\sim r_{\mathrm{S}}$. Hence, contrary to the collapse, the energy barrier of the escape remains finite as long as the skyrmion solution remains stable, as the skyrmion always has finite size.

All effects described above, such as the crossover, can be expected to be general features of skyrmion systems.

It is conjectured that this multi-scale approach, where an atomistic spin Hamiltonian is parametrised from first-principles electronic structure calculations and used in a statistical description of skyrmion stability, will give valuable insight into the foundational mechanisms determining the lifetime of skyrmions, as well as provide a tool for predictive materials design. Especially in the field of technological applications design, this may prove to reduce the efforts of searching for suitable materials compositions. Through the future usage of automation tools, such as AiiDa [161] or ASE [162], the human effort needed to screen material compositions and designs for applicability to proposed devices like the skyrmion racetrack will be greatly reduced.

Finally, note that the presented data may not be enough to gain a correct picture of how well a racetrack design would work, as only the previously well-known collapse and escape transitions were studied. It is presented in the following Chapter 2 how, using the saddle point search method presented there, an additional skyrmion transition was found, which may be highly relevant to racetracks on which multiple skyrmions are present.

SADDLE POINT SEARCHES 2

"Things are only impossible until they're not!"

— Jean Luc Picard

As previously discussed, the full set of transitions that a magnetic configuration can undergo or the final states are not necessarily known, but the knowledge of these states and transitions can be highly valuable. It has been laid out in the previous chapter that any transition, meaning the mechanism and the transition rate, is characterised by the corresponding first order saddle point in the energy landscape. This chapter will introduce methodology to seek out saddle points and will derive the necessary equations to transfer it to spin systems. The ability to *systematically* carry out such searches is applied to an accessible example as well as a skyrmion. Notably, this completes the class of methods for rate theory calculations for spin systems, which are formulated in a consistent mathematical framework and are now combined together in the *Spirit* framework, enabling new approaches and efficient exploration.

The main results presented in this chapter have been published by the author in Ref. [58].

2.1 MINIMUM MODE FOLLOWING METHOD

In the study of atomic rearrangements, such as chemical reactions and diffusion events, a category of methods for identifying possible transition mechanisms without knowledge of the final state has turned out to be highly useful [163, 164]. The category is that of single-ended saddle point search methods, which have been used extensively in studies of atomic rearrangements. As was previously alluded to, unexpected or unintuitive transitions can often turn out to be preferred over mechanisms that seem *a priori* most likely [165]. It can be expected that, though the context is quite different, the same may be found in spin systems.

The basic idea is quite simple: given a local minimum of the energy landscape as an initial state, the various possible transitions the system may undergo can be identified by climbing up the energy surface and converging on the various first order saddle points on the energy ridge surrounding the minimum.

The approach for this, discussed in this thesis, will be the minimum mode following (MMF) method [166, 167]. It is based on the curvatures of the energy landscape, which are the eigenvalues of the Hessian H.

As one starts out close to a local minimum, all curvatures of the energy landscape will be positive, i.e. the landscape is convex. The first step is to escape this convex region, for example by following a random vector, the gradient of the energy or even an eigenmode of the system.

Once an eigenvalue λ_i turns negative, one can follow the corresponding eigenmode Λ_i, by applying an effective force

$$F^{\mathrm{eff}} = F^{\mathrm{grad}} - 2(\Lambda_i \cdot F)\, \Lambda_i \,, \tag{2.1}$$

where $F^{\mathrm{grad}} = -\nabla\mathcal{H}$ is the negative gradient of the energy, i.e. the force for direct energy minimisation, and Λ_i is the normalized eigenvector corresponding to the negative eigenvalue. Note the notation, where these vectors are $3N$-dimensional for a system of N spins.

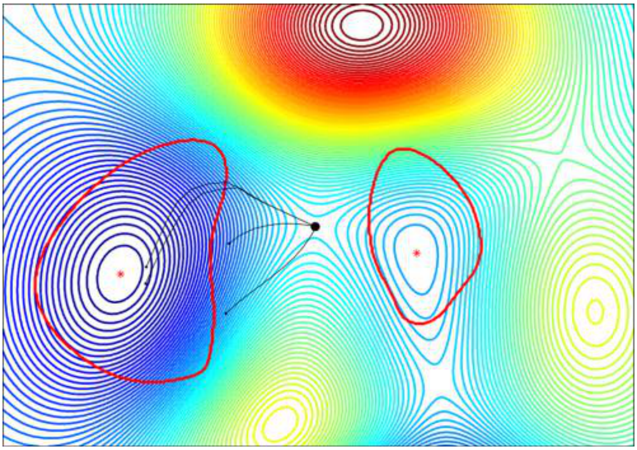


FIGURE 2.1 – MMF paths for atomic 2D system

A potential energy surface with minima (blue), maxima (yellow/red), and a saddle point (black dot). The black lines show saddle point searches starting from four different initial points. The red curves encircle the convex regions, i.e. they mark where the lowest eigenvalue is zero. See also Figure 2.4 for the corresponding figure for an atomistic spin system. Reprinted with permission from [167]. Copyright 2017 American Chemical Society.

While finite difference approaches can be used to determine the direction of negative curvature, the Hessian matrix H of an atomistic spin Hamiltonian can be derived analytically and the direction found through diagonalisation. As the system is displaced in the direction of the effective force, it moves to higher energy along Λ_i but to lower energy along the orthogonal degrees of freedom. Eventually, this should bring the system to a first order saddle point on the energy surface.

Note that if needed, the final state of the transition can be obtained by a slight displacement further along Λ_i, followed by energy minimization.

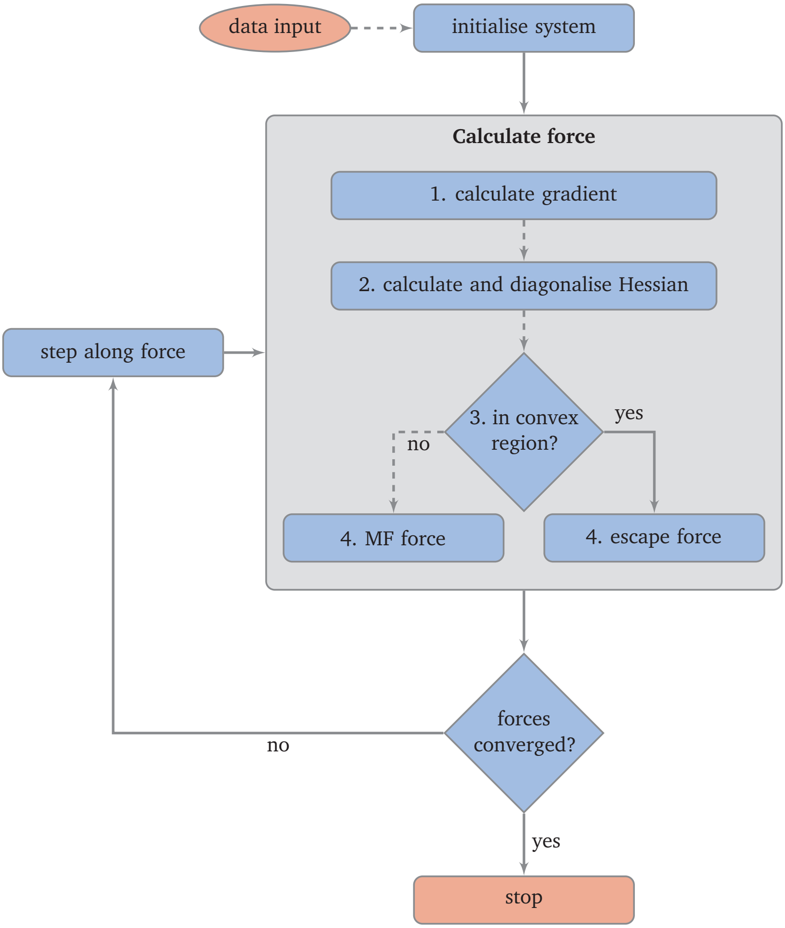


FIGURE 2.2 – Flow chart of the MMF algorithm

This flow chart visualises the general logic of the MMF method. After initialising the system, the iteration loop is started, in which the force is calculated and, if not yet converged, applied. The calculation of the force requires the calculation of the energy gradient and Hessian and the evaluation of the Hessian's eigenvalues to decide on how to proceed. If the system is in a convex region of the energy landscape, an escape force is applied (for example a random direction or the energy gradient) and otherwise the mode following force is used.

When applying this scheme, it is important to consider that the iterative optimization using $\vec{F}^{\text{eff}}$ may lead back into the convex region. If this leads to a back and forth oscillation it may significantly slow down convergence and, under unfavourable symmetries, even stall the calculation. Moreover, throughout a calculation, eigenvalues may cross, i.e. become degenerate, and the lowest eigenvalue mode may change significantly. In order to avoid unintentional switching between eigenmodes, one needs to distinguish different modes from each other. A straightforward solution is to take the scalar product of the current candidate and the previously used mode and, if a significant change in direction has occurred, search for the mode that has the (absolutely) largest scalar product with the previous mode. As long as iterations make small enough steps, this approach will work due to the fact that eigenmodes will change continuously while being mutually orthogonal.

In special cases the gradient may become orthogonal to the mode to within a given tolerance, in which case $\vec{F}^{\text{eff}}$ will simply produce a minimization in the gradient direction. Therefore, this case needs to be treated specially.

The following algorithm is implemented as the default behaviour in *Spirit* and is used in this thesis, where it is found to be successful:

- When the mode is not orthogonal to the gradient
  - if the mode has a negative eigenvalue: follow $\vec{F}^{\text{eff}}$
  - if the mode has a zero or positive eigenvalue: follow the mode.
- When the mode is orthogonal to the gradient
  - if the eigenvalue is zero (to within a tolerance): follow the gradient
  - if the eigenvalue is positive: follow $\vec{F}^{\text{eff}}$.

This algorithm is illustrated by Figure 2.2.

Analogous to the discussion of the nudged elastic band (NEB) method in Section 1.2, the above is held entirely general. The following Section 2.2 will discuss the special considerations needed for spin systems, which are not found in the literature discussing molecular systems.

2.2 CONSIDERING CONSTRAINTS

Let us recall Section I.2 of the Introduction and that the configuration manifold of an atomistic spin system with fixed spin lengths is a curved manifold embedded in Euclidean space $\mathcal{M}^{\text{phys}} \subset \mathcal{E}$. For the saddle point search method outlined in the previous section, the Hessian – the matrix of second derivatives – is needed for the calculation of eigenvalues and eigenvectors. The physical manifold is given by Equation (I.8):

$$\mathcal{M}^{\text{phys}} = \bigotimes_{i}^{N} \mathbb{S}^2 \subset \mathbb{R}^{3N}$$

and is a Riemannian manifold. As is well-known, second derivatives do not in general have an intrinsic geometrical meaning and therefore have to be treated with special care [87, 88].

It was mentioned in Section 1.3 that the Hessian can be calculated quite easily at stationary points, where $\partial\mathcal{H} = 0$. However, the MMF method requires its evaluation at arbitrary points of the manifold, requiring the evaluation of covariant derivatives. While the covariant derivatives on a spherical manifold are well-known in spherical coordinates, they suffer from singularities at the poles, where the coordinate system becomes undefined. The author therefore chose to use a $3N$ cartesian representation of the spin coordinates and represent the covariant derivatives in the same.

The Hamiltonian $\mathcal{H}$ can be smoothly extended to a function $\bar{\mathcal{H}}$ which is defined on $\mathcal{E}$. As discussed in Section I.2, the derivatives of the unconstrained $\bar{\mathcal{H}}$ are readily evaluated. We denote the gradient taken in the embedding space $\mathcal{E}$ as $\partial\bar{\mathcal{H}}$ and the gradient taken on the manifold $\mathcal{M}^{\text{phys}}$ as

$$\partial\mathcal{H} = P_x \partial\bar{\mathcal{H}}\,, \tag{2.2}$$

which is the projection of the gradient onto the tangent space at a point $x \in \mathcal{M}^{\text{phys}}$. The matrix of second derivatives in the embedding space is denoted $\partial^2\bar{\mathcal{H}}$.

Note that both the Introduction and Chapter 1 already made use of the fact that the first covariant derivatives are simply the derivatives in the embedding space, projected onto the tangent space of $\mathcal{M}^{\text{phys}}$.

In order to calculate the second derivatives on the manifold $\mathcal{M}^{\text{phys}}$, a projector-based approach, described in Ref. [89], is used. For any scalar function f on $\mathcal{M}^{\text{phys}}$ the true, covariant Hessian is defined as

$$\text{Hess}\, f(x)[z] = P_x \partial^2 \bar{f}(x) z + W_x(z, P_x^{\perp} \partial \bar{f})\,. \tag{2.3}$$

W_x denotes the so-called Weingarten map (see also Ref. [168]). For any vector v at a point x of a spherical manifold, the Weingarten map is defined as

$$W_x(z, v) = -z x^T v\,, \tag{2.4}$$

where z is a tangent vector to the sphere at x. In order to calculate the Hessian, $v = P_x^{\perp}\partial\bar{\mathcal{H}}$ is inserted to give

$$W_x(z, P_x^{\perp}\partial\bar{\mathcal{H}}) = -z x^T P_x^{\perp} \partial\bar{\mathcal{H}} = -z x^T x x^T \partial\bar{\mathcal{H}} = -z x^T \partial\bar{\mathcal{H}}\,, \tag{2.5}$$

where $x^T\partial\bar{\mathcal{H}}$ is the scalar product of the spin with the gradient.

In *Spirit* [47, 48], the Hessian is implemented in matrix representation, so in order to denote objects as they are implemented, we switch notation and drop the subscript x. For spin indices i and j, we write the gradient $\partial\bar{\mathcal{H}}$ as $\nabla_i\bar{\mathcal{H}}$ (now to be understood as a three-dimensional object) and the second derivative $\partial^2\bar{\mathcal{H}}$ as $\bar{H}$. Equation (2.3) within the Euclidean representation is then a $3N \times 3N$ matrix

$$H|^{3N} = (H_{ij}|^{3N}) = \begin{pmatrix} H_{11}|^{3N} & H_{12}|^{3N} & \cdots \\ H_{21}|^{3N} & H_{22}|^{3N} & \cdots \\ \vdots & \vdots & \ddots \end{pmatrix}, \tag{2.6}$$

which consists of N^2 blocks, each corresponding to a different spin-spin subspace. This matrix representation of the Hessian is obtained by acting with Equation (2.3) on the Euclidean basis vectors of the embedding space $\mathcal{E}$. Here, the subspace matrices of size 3×3 are defined as

$$H_{ij}|^{3N} = P_i\bar{H}_{ij} - \delta_{ij} I \vec{n}_j \cdot \nabla_j \bar{\mathcal{H}}, \tag{2.7}$$

where I denotes the 3×3 unit matrix and $\vec{n}_j$ the normalized direction of spin j. The resulting matrix $H|^{3N}$ will, however, have N eigenvectors orthogonal to the tangent space of $\mathcal{M}^{\text{phys}}$, representing the unphysical degrees of freedom in the embedding space $\mathcal{E}$. They can be removed by transforming the matrix using the tangent basis to $M \in \mathcal{M}^{\text{phys}}$, writing $H_{ij} = T_i^T H_{ij}|^{3N} T_j$, where T_i is the basis transformation matrix of spin $\vec{n}_i$ fulfilling $T^T P = T^T$ and $T^T T = I|^{2N}$. Thereby, the true Hessian of Equation (2.3), in the $2N \times 2N$ matrix representation, becomes $H = (H_{ij})$ with the spin-spin blocks defined as

$$H_{ij} = T_i^T \bar{H}_{ij} T_j - T_i^T I(\vec{n}_j \cdot \nabla_j \bar{\mathcal{H}}) T_j, \tag{2.8}$$

and which now contains only the physical degrees of freedom. From this matrix, the eigenmodes $\Lambda|^{2N}$ are calculated. The $3N$ representation is obtained as $\Lambda|^{3N} = T\Lambda|^{2N}$.

The basis transformation matrix T_i can be a 3×2 matrix of two tangent vectors to spin i, which can even be obtained from any random vector and another vector found by orthogonalization. For the implementation in *Spirit* the unit vectors of spherical coordinates θ and φ were chosen

$$T = \{\vec{e}_\theta, \vec{e}_\varphi\} = \begin{pmatrix} \cos\theta\cos\varphi & -\sin\varphi \\ \cos\theta\sin\varphi & \cos\varphi \\ -\sin\theta & 0 \end{pmatrix} = \begin{pmatrix} zx/r_{xy} & -y/r_{xy} \\ zy/r_{xy} & x/r_{xy} \\ -r_{xy} & 0 \end{pmatrix}, \tag{2.9}$$

where $r_{xy} = \sin\theta = \sqrt{1-z^2}$. Note that the poles of the sphere need to be excluded, but one may simply choose $\vec{e}_x$ and $\vec{e}_y$ and orthogonalize them with respect to the spin vector $\vec{n}_i$.

2.2.1 *Eigenmodes of the Hessian matrix*

The eigenmodes of the Hessian matrix are simply excitations along eigendirections of the energy landscape and we can dynamically visualise them by periodically applying them in the form of rotations to the spins.

$$n(t) \approx n(t_0) + c\Lambda_i \sin(\omega t), \tag{2.10}$$

where the amplitude c and frequency ω can be chosen arbitrarily in order to adequately visualise the mode (see also [40]). It should be clear that these eigenmodes have no relation to precession or damping, as the eigenmodes of the dynamical matrix do (see Section 1.5).

In Figure 2.3, this is visualised by the isosurface for $n_z = 0$, shown at different points of the dynamical excitation.

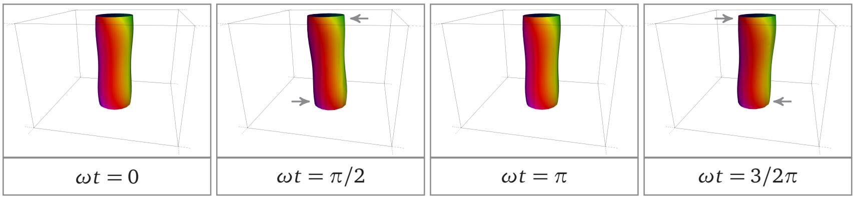


FIGURE 2.3 – Eigenmode of a skyrmion tube

3D skyrmion tube Hessian eigenmode visualised by the moving isosurface in *Spirit*, shown as four screenshots at different points in time. Arrows have been added to highlight the movement of the tube. This figure shows that eigenmodes, even in a 3D system, can be calculated and are intuitive and easy to visualise in *Spirit*.

Note the ease with which such complex eigenmodes can be understood in *Spirit*. It needs to be stressed that the direct access to the isosurface(s) and parameters, such as the amplitude for the application of the eigenmode, combined with the ability to view the system from any position and angle, make the process of analysing the resulting eigenmodes orders of magnitude faster. These kinds of improvements in productivity for the user are the strength points of *Spirit*

2.3 MMF APPLIED TO A SINGLE SPIN

The first example will be – as in the first example of the GNEB method – a single spin in a gaussian superposition potential. We recall Equation (1.17):

$$H = \sum_i H_i = \sum_i a_i \exp\left(-\frac{l_i^2(\vec{n})}{2\sigma_i^2}\right),$$

where a_i is the amplitude, σ_i the width and $l_i(\vec{n})$ is a measure of distance of the spin $\vec{n}$ to the center $\vec{c}_i$ of the ith gaussian function. Parameters for the test case are given in Table 2.1, which are a slight variation on the parameters of the GNEB example, given in Table 1.1. See Appendix B for the corresponding derivatives, needed for this method, as well as the Python script with which the saddle point searches were performed.

a	σ	p_x	p_y	p_z
0.5	0.2	−1	0	0
0.7	0.25	0	0	−1
−0.8	0.1	1	0	−0.6
−0.8	$5 \cdot 10^{-2}$	0.8	0	0.5
−0.5	0.2	0	1	0
−1	0.2	0	−1	0
−1	0.1	0	0	1

TABLE 2.1 – Parameters of Gaussian Hamiltonian

These parameters are chosen to provide sufficiently interesting features in the energy landscape, with a central minimum from which to initiate the calculations and three saddle points in the vicinity.

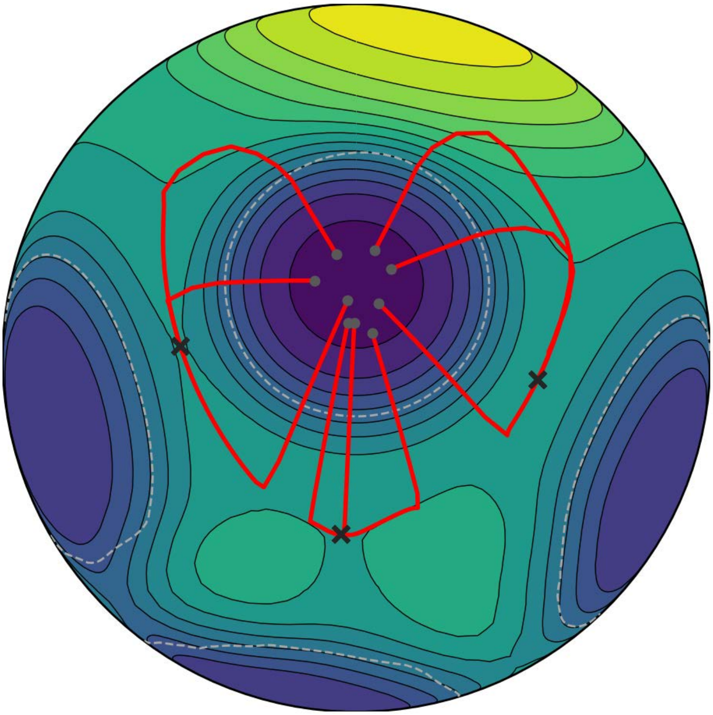


FIGURE 2.4 – Mode following path of a single spin

An example of a saddle point search for a single spin in a Gaussian superposition potential. In this case, the energy surface can be mapped onto a sphere. Saddle points are marked by black ×. The red curves are the saddle point search paths starting from various points close to a selected local minimum. In this case, the energy gradient is chosen as a force vector to escape the convex region (encircled by the white dashed line), and the minimum mode is followed thereafter. The three reachable saddle points are found. This figure has been published in [58]. Copyright 2018 American Physical Society.

It turns out that the eigenmodes of the single spin in this potential are largely unuseful, as they often lead to circular motions around the local minimum (see Figure 2.5). Consequentially, the gradient was used to escape the convex region. This yields good results for this system due to the symmetric nature of the potential around the minima and the low-dimensional parameter space.

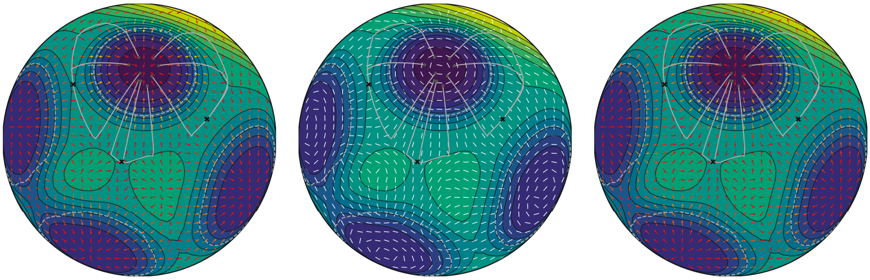


FIGURE 2.5 – Spherical plot of the mode following force contributions

The vector fields of the gradient force (left), minimum mode (middle) and the resulting effective force (right) are shown. The gradient force points away from maxima, towards minima of the energy landscape. The minimum mode represents the symmetry of the energy landscape and, close to the saddle points, points in the direction of the degree of freedom corresponding to the transition between two of the minima. The effective force, inside the convex region (encircled by dashed gray lines), is simply the gradient, therefore pointing radially outwards from the minima; inside the remaining regions, it is the minimum mode following force. One can clearly see the effective force pointing towards the saddle points. These images are shown in large in Appendix D.

In order to make sure that the Heisenberg Hamiltonian is working correctly, one can do analogous tests for an interacting spin system, where one spin is pinned (e.g. to the z-direction). For the following example, parameters are chosen to be

$$K = 4\,, \quad J = 1\,, \quad \vec{D} = (0,0,1)\,. \tag{2.11}$$

In this case it suffices to simply zero the forces applied to the pinned spin and to only plot those forces and modes belonging to the movable spin, as shown in Figure 2.6. It should be noted, however, that in a more general case one should also take the Hessian matrix into consideration: if one sets the blocks of the pinned spins to zero, one creates zero modes, but setting them to very large values (like a very narrow and steep minimum for the spin to reside in) can cause numerical instabilities. As Figure 2.6 illustrates, the minimum mode clearly reflects the symmetries of the underlying energy landscape and lets us invert approximately the component of the gradient force orthogonal to the energy isolines. The resulting force field therefore gives a straightforward convergence onto the saddle point.

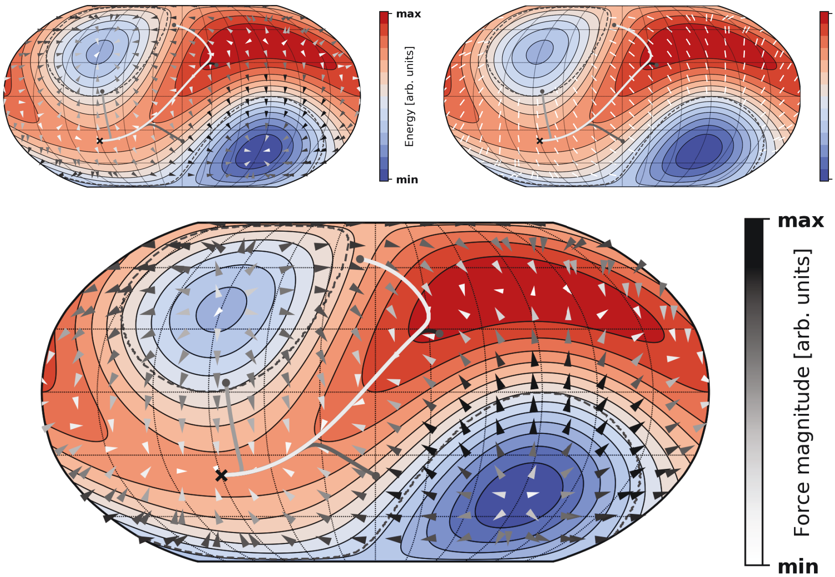


FIGURE 2.6 – Mode following forces of a two spin system

With the second spin pinned, we can again plot everything on a sphere. In this case, it is more useful to use a complete projection (here the so-called "Robinson" projection is used). As in Figure 2.5, the effective force clearly directs the spin to the saddle point. These images have been published in [47].

2.4 NOVEL SKYRMION TRANSITION REVEALED BY THE MMF METHOD

After the successful test of the MMF method for simple test systems with only one and two spins, the logical next step is to apply it to a more complicated case, such as an isolated skyrmion. We recall the Heisenberg model (I.1), where for this section a slightly simplified version is used:

$$\mathcal{H} = -\mu_S \sum_{i=1}^{N} \vec{B} \cdot \vec{n}_i - J \sum_{\langle ij \rangle} \vec{n}_i \cdot \vec{n}_j - D \sum_{\langle ij \rangle} \vec{d}_{ij} \cdot (\vec{n}_i \times \vec{n}_j), \tag{2.12}$$

where $\vec{B}$ is a uniform external magnetic field, J is the nearest neighbour exchange coupling constant, D the nearest neighbour Dzyaloshinskii-Moriya interaction (DMI) constant and the unit vectors $\vec{d}_{ij}$ are the Dzyaloshinskii-Moriya unit vectors in the plane of the lattice parallel to the vector connecting spins i and j. The system consists of 40×40 spins on a square lattice with open boundary conditions and the parameter values and field strength are chosen to be the same as in Ref. [136], where $D = 0.45\,J$

and $H = 0.8\,H_\mathrm{D}$, where $H_\mathrm{D} = D^2/(\mu_\mathrm{S} J)$ is the critical field strength. For the parameter values used here and choosing $J = 1$ meV, we get $H_\mathrm{D} = 3.5$ T.

The Bloch skyrmion, shown in Figure 2.7, is a local minimum of the energy landscape and therefore metastable with respect to the ferromagnetic phase. Figure 2.7 also shows an illustration of the eigenvectors corresponding to the three lowest eigenvalues, corresponding to translation, breathing and elliptical distortion.

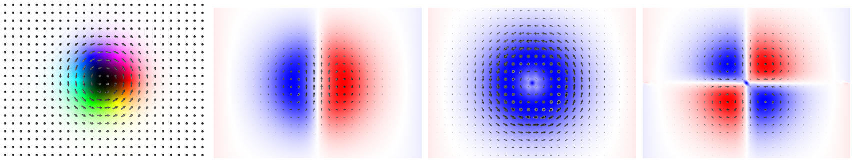


FIGURE 2.7 – Lowest three eigenmodes of an isolated skyrmion

From left to right: isolated skyrmion, the translational, breathing and fourfold – or elliptical – eigenmodes of the isolated skyrmion. While there are two translational modes with eigenvalue zero, the breathing mode and the elliptical mode correspond to a positive curvature. These images have been published in [58]. Copyright 2018 American Physical Society.

For comparison see also the analysis of the dynamical skyrmion eigenmodes presented in Ref. [169].

As mentioned in Section 2.1, there are several approaches to escape the convex region of the energy landscape, where arguably the simplest approach is to follow a random direction until a negative eigenvalue is found. For the skyrmion it turns out that it is highly effective to follow a selected mode right from the beginning. After the corresponding eigenvalue turns negative, the effective force given by Equation (2.1) is used, where Λ_i is now the eigenvector of the initially selected mode rather than the one with lowest eigenvalue. Looking at the images in Figure 2.7, it should be clear why this can lead the system straight to a saddle point. The translational mode, which initially has the lowest eigenvalue, will move the skyrmion towards the boundary of the system, until the mode for the escape is found and eventually, as it is pushed along the effective force, it converges onto the saddle point corresponding to the escape through the boundary. A small displacement along the unstable mode and subsequent relaxation of the system will then bring it to the ferromagnetic state. By following the breathing mode, the skyrmion can shrink until the eigenmode of the radial collapse is found and the system converges to the corresponding saddle point. It turns out, however, that the elliptical eigenmode leads to a stretching of the skyrmion until its center becomes energetically unfavourable and the mode turns into a bisection of the skyrmion in the middle. This brings the system to the saddle point of a skyrmion duplication transition, which has not been covered in the literature so far. The saddle points, which were found by this approach, are shown in Figure 2.8.

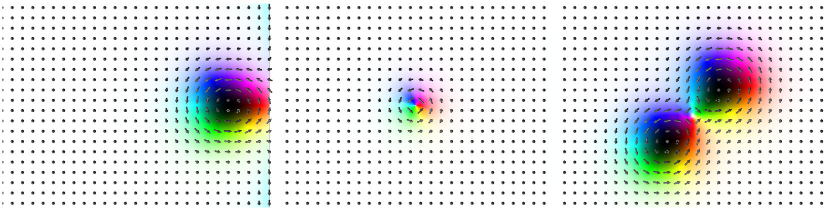


FIGURE 2.8 – Saddle points of three skyrmion transitions: escape, collapse and duplication

From left to right: saddle points of the skyrmions escape through the boundary, the radial collapse and the duplication transition. Note that the duplication transition can, inversely, be viewed as a skyrmion collision transition, which may under certain conditions be an important considerations for skyrmion racetracks. These images have been published in [58]. Copyright 2018 American Physical Society.

Due to the fact that random directions, here, may quickly lead to negative eigenvalues belonging to collective modes, the approach of following the skyrmion's modes was more successful.

The energy along the minimum energy paths (MEPs) for the three transitions is shown in Figure 2.9. These were calculated using the GNEB method, using an initial path formed by linear interpolation between the intial state and the saddle point, as well as between the saddle point and the final state.

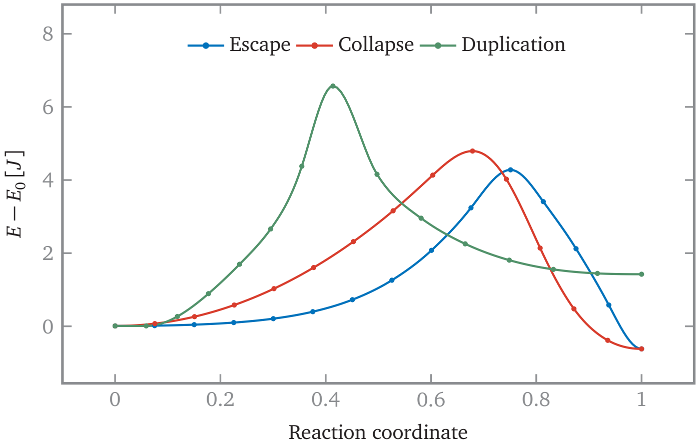


FIGURE 2.9 – Minimum energy paths of the escape, collapse and duplication transitions

Minimum energy paths for the three types of transitions found: Duplication, collapse and escape. The reaction coordinate is the scaled total displacement along the path. The energy is given in units of the exchange coupling constant, J. This plot has been published in [58]. Copyright 2018 American Physical Society.

The energy barrier for the collapse and escape is quite similar, while in this case the duplication has a higher barrier and a noticeably different shape. The total displacement along the MEP from the initial state to the saddle point is shortest for the duplication, while the saddle point for escape involves a larger and the displacement of the skyrmion to the boundary the largest absolute displacement (where the latter of course depends on the system size). The collapse has the expected profile, as previously reported [56, 170]. Note that the duplication of the skyrmion leads to an increase in the energy of the system because the skyrmion is only metastable with respect to the ferromagnetic state for this set of parameters. The other two transitions reduce the number of skyrmions in the system and therefore lead to a decrease in the energy.

Naturally, the relative height of the energy barriers for the three processes depends on the choice of parameters. Using a 10% smaller field and 60% larger value of D/J than in the previous example gives a lower activation energy for the duplication than for the other two transitions. This is shown in Figure 2.10.

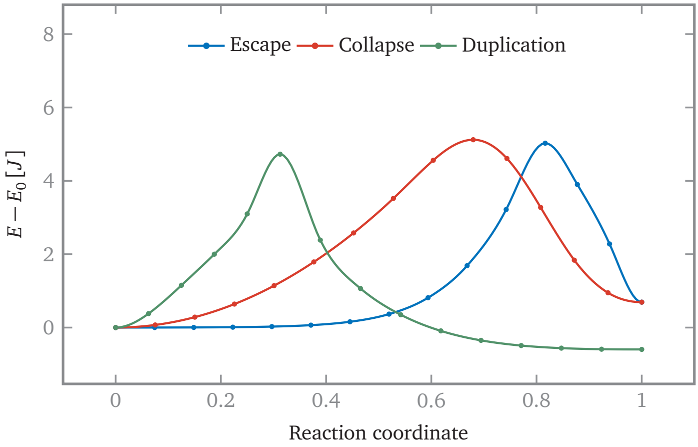


FIGURE 2.10 – Minimum energy paths of the escape, collapse and duplication transitions at different parameters

The energy barriers at a second parameter set, where the field is 10% smaller and the value of D/J 60% is larger. This shows that the energy barrier for the duplication and collision can be of the same order of magnitude as the boundary escape/nucleation and the radial collapse/nucleation. This plot has been published in [58]. Copyright 2018 American Physical Society.

The duplication transition identified in the context of this work has not been described previously, but has further features of interest. It may turn out to be an important mechanism for generating skyrmions. Past its saddle point, the MEP of the skyrmion duplication contains important information about skyrmion interaction, as it shows how the repulsive interaction potential between skyrmions varies with the distance between them. Also, when viewing the transition inversely, it represents the merging, or collision, of two skyrmions to form a single skyrmion. This could turn out to be an important consideration for the design of skyrmion racetracks.

In this context, the question arises whether it is possible to dynamically induce duplication events. Starting from the fact that the duplication transition is found by an elliptical deformation of the skyrmion, it is reasonable to try and excite this kind of mode in a dynamical simulation. To achieve the elliptical skyrmion deformation, an inclined, short magnetic pulse is applied. As there is a constant external magnetic field, for the duration of the pulse – 200 ps – the total magnetic field is tilted. Magnetic field pulses are known to be an efficient way of exciting nonlinear skyrmion dynamics [142] and the results, shown in Figure 2.11, support the idea.

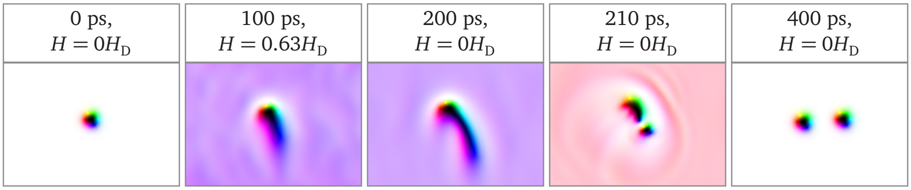


FIGURE 2.11 – Dynamics simulation of skyrmion duplication

Snapshots from a dynamical simulation where the duplication of a skyrmion is induced by applying a magnetic pulse over 200 ps, giving a total field that is tilted with respect to the normal of the plane. The labels on top of the frames give the time and magnetic field of the pulse (which adds to a constant field in the direction of the normal to the plane). The pulse is applied at time t=0 and lasts until $t = 200$ ps. At the end of the pulse, an elongated defect is formed which later splits up (at $t = 209$ ps) to form two skyrmions. These images have been published in [58]. Copyright 2018 American Physical Society.

The pulse has a magnetic field directed along the vector $\{x,z\} = \{0.8,-0.61\}$ while the static magnetic field is directed along the normal vector $\{x,z\} = \{0,1\}$ so the pulse partly cancels out the static field. The static field has magnitude $H = 0.8\ H_D$ but during the pulse the total magnetic field drops to $H_p = 0.63\ H_D$. The time evolution was carried out using the semi-implicit midpoint algorithm of Mentink *et al.* [118] with a time step of 0.01 ps.

While the out of plane component of the pulse brings the skyrmion closer towards elliptical instability, the in plane component triggers the skyrmion to elongate and, over time, form into a U shape. It therefore represents a combination of different eigenmodes of the skyrmion in its ground and elongated state. When the pulse is switched off again, the excitations (or spin waves), which then move along the elongated skyrmion and cause it to split in half.

Other, more efficient or more reliable ways of inducing this duplication could probably be devised and the presented results should be taken as a proof of concept for the existence of this novel transition – expected to be a quite general phenomenon.

Suggestions are to pin the skyrmion, to introduce defects or by more complicated external stimuli.

An argument in favour of the generality of the results is the presence of the first order saddle point for parameters of a Pd/Fe bilayer on an Ir(111) substrate, a system which has been extensively studied, both experimentally and theoretically, with good agreement between the used parameters and experiment [38, 171]. In this system, the energy barrier for the duplication turns out to be slightly lower than that of the collapse, namely 78 meV versus 80 meV. In a follow-up study, performed by two students under the supervision of Prof. Jónsson and the author, it was investigated how prevalent the duplication transition is, i.e. in which parameter ranges of magnetic field, D/J and K/J the saddle point of the duplication is of first order. The lowest eigenvalues at the saddle point of the transition were evaluated in order to ensure that the saddle point is of first order and to determine the points where the lowest eigenvalue converges to zero and the path can no longer be relaxed by the GNEB method. The results are shown in Figure 2.12.

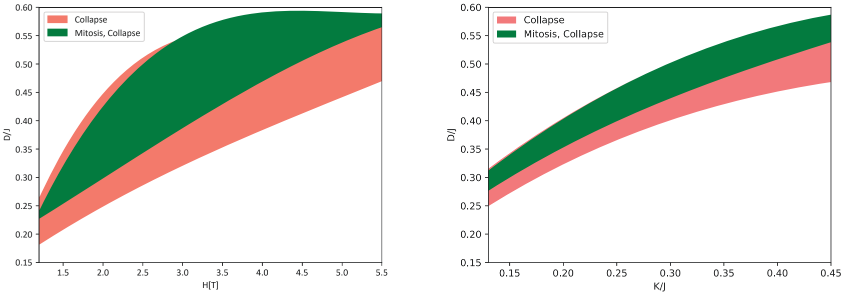


FIGURE 2.12 – Parameter ranges where the skyrmion duplication is a minimum energy path

The parameter range of external field H (left) and anisotropy K (right) over DMI interaction D in which the collapse transition (red) and mitosis (green) exhibit a first order saddle point. In order to keep the generality of the results, parameters are given as ratio over the exchange constant J.

The skyrmion mitosis transition is found to be a stable minimum energy path in a wide range of parameters, meaning that for many systems it may turn out to be a relevant transition, influencing skyrmion stability or posing opportunities for the creation of skyrmions. Evidently, larger values of DMI support this transition, which fits together with the fact that the saddle point includes a small region of strong curl in the magnetisation, somewhat like a defect.

These results warrant further investigation still, as it is not clear how the prefactor Γ_0 of this transition behaves in dependence on the shown parameters. Estimation of the rate of duplication and comparison with that of the other mechanisms as a function of parameter values remain interesting future tasks.

3 IMPLEMENTATION: SPIRIT - A SPIN SIMULATION FRAMEWORK

"Es ist unwürdig, die Zeit von hervorragenden Leuten mit knechtischen
Rechenarbeiten zu verschwenden, weil bei Einsatz einer Maschine
auch der Einfältigste die Ergebnisse sicher hinschreiben kann."

— Gottfried Wilhelm Leibniz

As mentioned repeatedly throughout this work, at the outset of the project, there was a significant lack of modern tools for atomistic spin calculations – none of them providing a graphical user interface (GUI) for real time visualisation or direct access to calculation parameters. In order to achieve these benefits, as well as optional parallelisation on multi-core central processing units (CPUs) and graphics processing units (GPUs), C++ was chosen as the programming language in which the software should be implemented. For these reasons, the *Spirit* software was created – a specific set of goals in mind:

- increase productivity of scientists running spin simulations
- reduce the barrier of entry into the field and ease understanding
- unify interfaces and methods of this field

The approaches described in the Introduction and Chapter 1 have been collectively implemented into a library forming the core of the *Spirit* Framework. This chapter will describe and motivate the code's general design and structure and give additional detail on how the methods and solvers were implemented numerically, and how the author attempted to provide flexibility without loss of function. Furthermore, this chapter will try to elaborate on the challenges in the layout of a software project of this scale, explain the rationale behind certain decisions and suggest how to overcome them. Significant aspects of the descriptions and verifications presented in this chapter have been published by the author in Ref. [47].[1]

[1] The paper is on arxiv and is under review at Physical Review B

Spirit [47, 48] is an open-source project and has been published to `GitHub.com`.[2] It should be noted that several other services, integrated with the hosting service, were employed to automate building and testing of the framework on various platforms (called continuous integration) – a necessary approach in a project of the given scale and generally recommendable.

3.1 CODE STRUCTURE

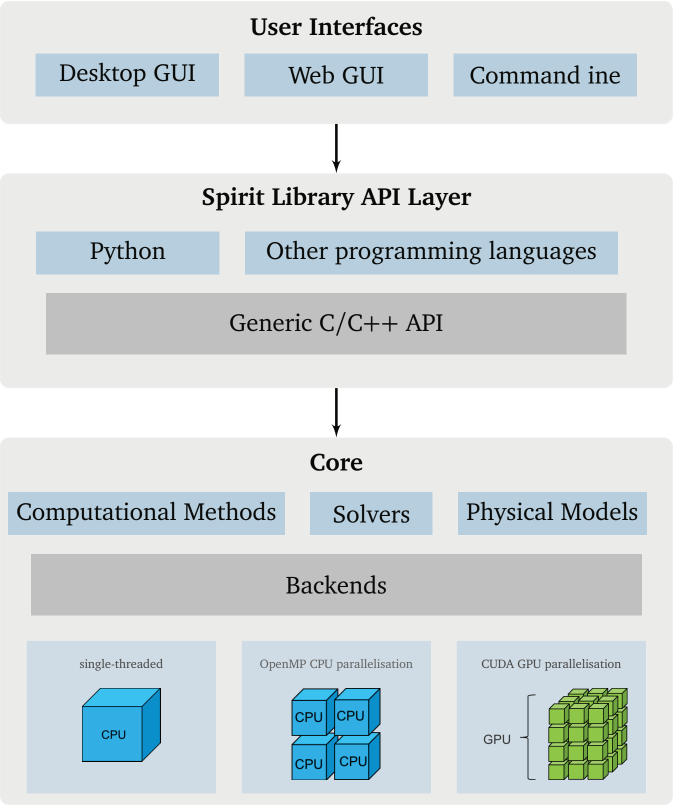


FIGURE 3.1 – Schematic of *Spirit* code structure

This flow chart visualises how the code is vertically layered. By placing an application programming interface (API) layer on top of the core library, it can be used generically and from various languages. Using this API for the computations, data manipulation and input/output, anyone is free to implement other or interchangeable user interfaces. This figure has been adapted from [47].

[2] `spirit-code.github.io`

As illustrated in Figure 3.1, the framework is divided into three main layers: I) the core library, containing all physical simulation methods II) an API layer, available in several programming languages III) several user interfaces with which to interact with *Spirit* more directly.

The core of the *Spirit* software is a library, which comprises the methods described in the previous chapters, the energy minimizer and dynamical solvers, the Heisenberg Hamiltonian, input and output. This structure is visualised in Figure 3.4.

It is worth noting the backends, which abstract the lower level performance-critical routines, and which are used throughout the code. Figure 3.2 shows an examplary speedup, which can be obtained using the CPU and GPU parallelisations for a cube of 100 × 100 × 100 spins with nearest-neighbour exchange, DMI and dipolar interactions.

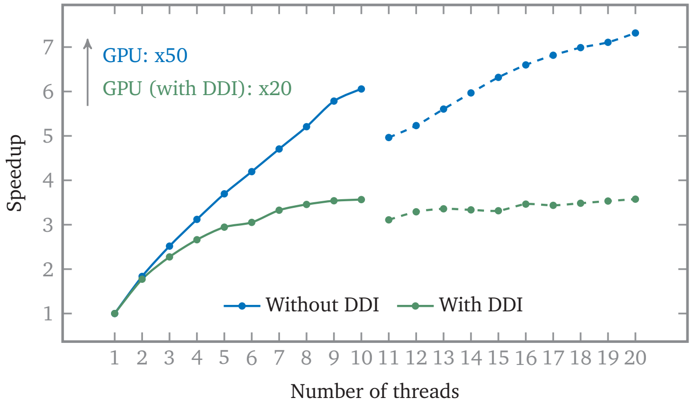


FIGURE 3.2 – Performance scaling of *Spirit* over number of parallel threads

An example performance measurement for *Spirit* is shown for a system of size 100 × 100 × 100 with nearest neighbour interactions with and without dipolar interactions. The speedup on the 10-core CPU scales well with the number of cores (solid lines). When using hyperthreading (dashed lines) the speedup at 20 threads on 10 cores is higher than without hyperthreading, but only by a factor of ~ 1.2 at twice the number of threads. The case with dipolar interactions scales worse and seems to be limited by the calculation of the FFTs, which cannot be trivially parallelized. In both cases, the GPU gives a significantly better speedup. As a CPU, the Intel Core i9-7900X 3.30GHz was used and the NVIDIA GeForce GTX 1080 was chosen for the GPU. Note, however, that the scaling behaviour may depend heavily on system size and number of atoms in the basis cell, number of neighbour shells, solver and method used.

Through this separation, the amount of code written for parallelisation is significantly reduced, as the more abstract code does not need to be manually parallelised, but can instead use the backend functions without any need to care about details of performance on that level.

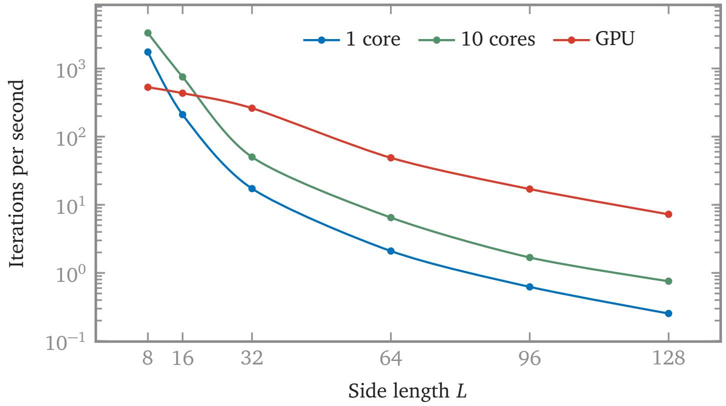


FIGURE 3.3 – Performance scaling of *Spirit* over system size

The speed of a LLG simulation for a simple cubic lattice of varying size with external field, nearest-neighbour exchange and DMI and dipolar interactions. The calculation without dipolar interactions on one core is very close in speed to the calculation with dipolar interactions on 10 cores. This figure has been published in [47].

The core library has an application programming interface (API), written in the C programming language, which gives access to its features through function calls. As many other programming languages can directly use C functions, this significantly increases the number of potential usage scenarios for this library. Two example usage scenarios, which were realised in the context of this work, are the Python bindings and the graphical user interface (GUI).

A notable feature of the core library is its logging, where initial input, relevant user actions and output are written to the console, as well as optionally saved to a log file. These log messages can be used for initial checks of consistency, such as the version of spirit used and initial parameters (this can also greatly improve reproducibility), to monitor running calculations, to extract performance timings and potentially to obtain debugging information when something goes wrong.

The desktop GUI of Spirit was written in C++ using the QT framework [172] in order to be able to provide a performant cross-platform interface into which the core library is tightly integrated. A visualisation library was written by Florian Rhiem for the purpose of high-quality real-time visualisations in this GUI. Aside from these visualisations, which are shown throughout this thesis, the main goal of the GUI is to enable direct interaction with simulations. Some of these features are shown in Figure 3.5.

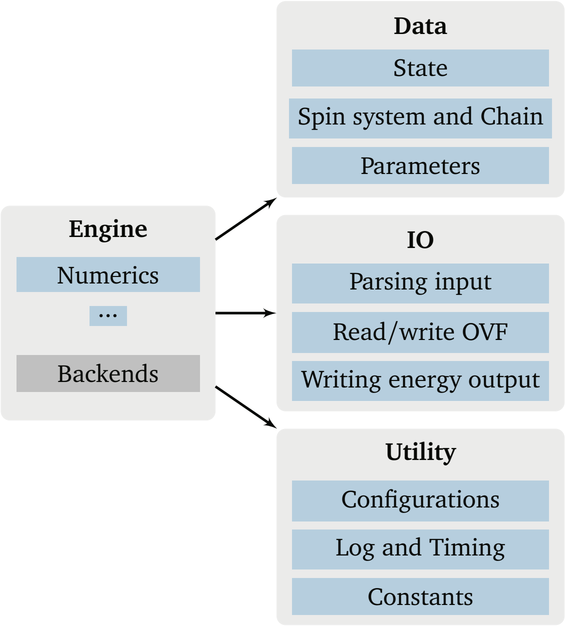
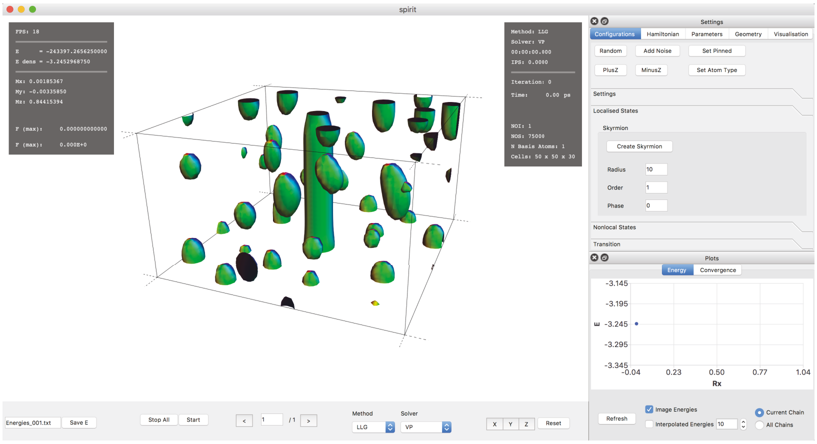


FIGURE 3.4 – Schematic of *Spirit* core library layout

This schematic depiction of the structure of the core library shows how the code is modularised, in order to avoid strong coupling between code components. This means that rewriting, for example, the OVF file parser, will not require significant changes in other areas of the code.



```
FPS: 18

E      = -243397.265625000
E dens = -3.2452968750

Mx: 0.00185367
My: -0.00335850
Mz: 0.84415394

F (max):     0.000000000000
F (max):     0.000E+0
```

```
Method: LLG
Solver: VP
00:00:00.000
IPS: 0.0000

Iteration: 0
Time:      0.00 ps

NOI: 1
NOS: 75000
N Basis Atoms: 1
Cells: 50 x 50 x 30
```

Radius 10
Order 1
Phase 0

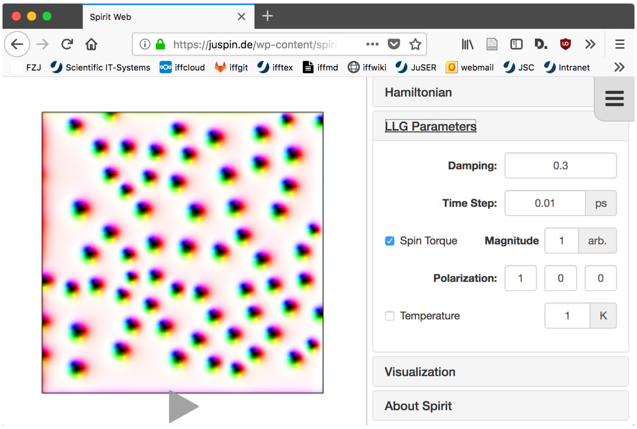
Energies_001.txt
1 / 1
Method LLG
Solver VP
☑ Image Energies
□ Interpolated Energies 10
● Current Chain
○ All Chains

FIGURE 3.5 – Screenshot of *Spirit* graphical user interface (GUI)

The GUI of *Spirit* is shown with the visualisation of a complex 3D magnetic texture using isosurfaces, informational widgets and parameter control interfaces.

Through its C API, the core library can be used to program command line tools, for example for use on clusters, but the more versatile approach is to use the Python bindings and write scripts instead. The latter gives very easy access to, for example, the file system and many libraries for post-processing, e.g. to precisely track skyrmion positions over the course of a simulation.

At time of writing, an unusual feat for a scientific code, accomplished through this code structure, has been the creation of a web-based user interface, which can be run in any current internet browser, and with which atomistic spin dynamics simulations can be run on mobile devices, such as smartphones. This is achieved, similar to the Python bindings, by calling the C library API, in this case from javascript code run inside a website. The visualisation was again created by Florian Rhiem. The Web interface gives a unique opportunity to make the code broadly available without any effort for installation. While – at the current stage of the technology – this means that the parallelisation features cannot be used and that performance is limited, it removes further barriers of entry and, through interaction with the parameters and visualisation, makes it easy to gain an intuitive understanding. The Web interface, as seen in a window of the Firefox web browser, is shown in Figure 3.6.



Damping: 0.3

Time Step: 0.01 ps

☑ Spin Torque Magnitude 1 arb.

Polarization: 1 0 0

□ Temperature 1 K

FIGURE 3.6 – Screenshot of *Spirit* Web user interface (UI)

The Web UI of *Spirit* is shown with the visualisation of a skyrmionic 2D magnetic texture using interpolated values on a 2D plane and parameter control interfaces.

For educational purposes, this interface could be a useful tool to teach students about the relations of different interactions and about the LLG equation and the dynamics it produces. It can also be used for presentation and could – if implemented into the website – even be used for data visualisation, e.g. when distributing calculation results between research groups. It should be stressed that in principle, all methods implemented in Spirit, such as GNEB or the visualisation of eigenmodes, could be used in the Web interface. As we now carry computers in our pockets with the computing power of supercomputers of not-so-long ago, this may well be the future way to perform atomistic spin calculations – at least for reasonable system sizes – quickly and easily and without the need for technical expertise.

3.1.1 *Considerations for the variety of methods*

The fact that the various methods implemented in *Spirit* can and should be used together means that a user needs control, at the same time, over the parameters of a single spin system and all methods that might run on it, as well as parameters of the collection (or chain) of systems and corresponding methods. It also implies that systems should be easy to add to or remove from the collection

There are several issues one can run into when designing a program for this, as each method has its own requirements, which need to be checked whenever the user changes certain parameters. For example: what should happen if a user sets several different geometries and tries to start a GNEB calculation? And how can one make it easy to set all geometries to be the same again?

In *Spirit*, the design choice was to allow any combination of calls to the API, but to log warning messages when the behaviour might have unintended consequences and to log error messages whenever an action is not possible. This means that all checks for consistency can be made and applicability of any action or calculation method can all be verified in the API layer, meaning they never have to be done inside the actual core library. In consequence, *Spirit* should only crash in rare events, such as running out of memory, but not due to user error, such as invalid input.

A difficulty in the layout of the core library was the reduction of duplicate code by finding the abstract commonalities of all the methods, from MC over GNEB to MMF. Mainly, this is the concept of iterating, where N iterations are made either through a single API call or N API calls, and functionality such as the calculation of how many iterations are being performed per second.

For several of the methods, namely LLG, GNEB and MMF, solvers are used to iteratively advance the spin configuration according to the given equations. This means that all solvers, such as Heun's method or the velocity projection method, should be useable for all of these methods. Here, the challenge is for the solver to correctly extract the required forces acting on the spin system(s) without knowledge of the method currently in use. In fact, the solver should be agnostic to almost everything except the parameters needed for its scheme of advancing the spin vectors to the next iteration.

3.1.2 *Considerations for parallelisation*

A highly important aspect of any code for numerical calculations is performance and, in consequence, parallelisation. However, a great variety of heterogeneous platforms exist, where a CPU can be single-threaded or multi-threaded and an accelerator, such as a GPU may be available. To complicate things, it is a current-day norm to execute code in parallel on multiple nodes of a computation cluster, increasing the potential size of problems by adding an additional layer of parallelization (note that each node might have a GPU) as visualised in Figure 3.7.

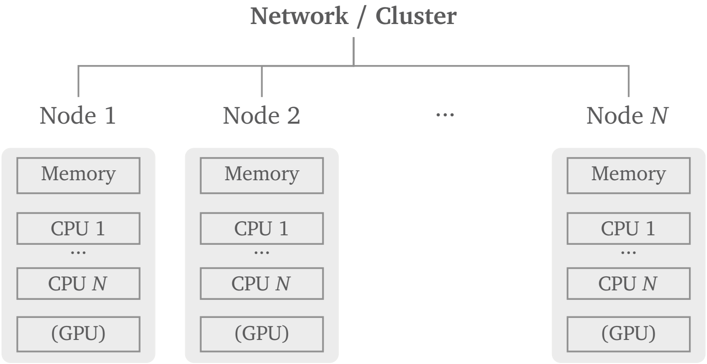


FIGURE 3.7 – Schematic of parallelisation over multiple nodes

Using, for example, the message passing interface (MPI), execution of a program can be distributed between multiple nodes, each with their own memory, CPU and potentially accelerators, such as a GPU. This form of parallelisation is not used in the core library of *Spirit*, as it was not found to be beneficial on the level of the implemented methods.

In *Spirit*, the potential complexity of various parallelisation layers was mitigated by two design decisions: I) parallelisation over multiple nodes should be done in specific applications (a problem-specific executable or Python script), not in the generic core library of *Spirit* II) for CPUs, the parallelisation on a single node is covered by OpenMP [173] and for GPUs it is performed by CUDA [174], where code duplication is minimized by the usage of *backends*, separated from the more abstract parts of the code, such as the solvers.

Spirit implements two backends (see Figure 3.1), both with the same interface of functions to call. The CPU code of the library is written only once and only the implementation is – in a sense – duplicated in the form of CUDA kernels for the numerically time-consuming functions. The clean separation of the heavy-lifting backends and the more abstract code allows the solvers and methods to be independent of the parallelisation and effectively orthogonalises the problems of functionality and performance.

An important technical aspect of the GPU backend is the usage of unified memory between CPU and GPU, meaning that the relevant objects are automatically created on both the host and target devices and memory is automatically transferred when needed. While this can have some negative performance impact, it greatly simplifies the programming and logic, reduces the required effort and has the benefit that no data is copied unless needed. This also enables better abstractions of the data containers.

While most of the backend of *Spirit* is hand-written for its specific use-case, there are computational and parallelisation libraries available, which give reasonable alternatives to the approach taken here. To the authors knowledge, the most relevant such library is *VexCL* [175], which also very effectively hides the parallelization when implementing mathematical equations.

Finally, there is an important issue in the parallelisation of the entirety of methods, namely that all those, which do not iterate some solver, cannot be trivially parallelised in the same way. The Metropolis Monte Carlo algorithm, for instance, relies on random sampling of the spins during an iteration, meaning that the system cannot be trivially parallelised. Instead, the system has to be subdivided into domains, which are sufficiently uncorrelated, in a kind of checkerboard pattern [176], as shown in Figure 3.8.

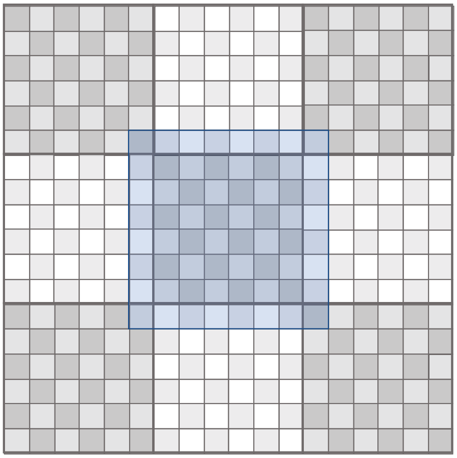


FIGURE 3.8 – Checkerboard domain decomposition for parallelised Monte Carlo

Double checkerboard decomposition of a 18 × 18 lattice, enabling parallelised Metropolis Monte Carlo simulations. Each of the 3 × 3 large tiles is assigned as one thread block, where individual threads of e.g. a CPU or GPU operate on one of the two sub-lattices of all 6 × 6 sites of the tile in parallel. The large tiles of the coarse decomposition can be updated independently. The blue shading indicates the overlap of the one large tile with its neighbours for the case of only nearest-neighbour interactions. Reprinted with minor modifications from [176] with permission from Elsevier.

Especially when long-ranged, e.g. dipolar interactions are involved, this can become increasingly difficult, though approaches have been developed for this as well [177]. The diagonalisations of the Hessian and the dynamical Matrix in the MMF and HTST methods, respectively, requires entirely different considerations, where most likely it is the best approach to use libraries, which have been written by experts for these specific tasks.

One might intuitively think it a good idea to parallelise the GNEB method on the level of the different images, iterating the images in parallel, for example using message passing interface (MPI) to iterate each image on a separate node. However, the author has found this to be not true in practice. The additional effort, needed to implement MPI calls, does not seem worth it when the only benefit is the faster execution on a large compute cluster. Usually, one will do significantly more than just one calculation, so that the parallelisation can instead (much easier) be performed on the level of executing multiple GNEB calculations.

3.1.3 *Considerations for simplicity*

A very common problem in the design of the layout of code for a software project is the question which parts should be modularised and which should be integrated. In most cases, there are various benefits to either side. In the case of *Spirit*, one of the main concerns was generality and extensibility to more general use cases than those covered in this work. Hence, the core library was turned into a module, which can be re-used in different contexts. Additionally, it was therefore crucial to make the APIs as easy to use as possible without restricting the parameters, which can be adjusted, or the combination of methods, which can be used together.

Note that every modularisation comes with the introduction of new interfaces to the interacting modules. It is therefore usually only ever worth the cost, if a component is reused. Otherwise it can significantly reduce simplicity.

Analogous to this is simplification by constraint. Such a constraint was initially used in the early development of *Spirit*, where it seemed sensible to have a constant geometry for the spins during one execution of the code, as it eliminated some corner cases to check and provided some potential performance improvements. The more constrained the use cases are, the more performant the corresponding code can be. It later turned out that this constraint hindered some use cases, such as the comparison of energy density for a spin spiral in different lattice sizes or the enlargement of the system to fit a state which turned out to be larger than expected, and it was very impractical to have to restart the code. Another example is that a lot had to be refactored in *Spirit* in order to accomodate the possibility of disordered distributions of atom types and atom-dependent moment lengths. The elimination of such constraints can be a significant effort and their implementation should therefore be carefully evaluated beforehand.

Another hurdle in the development of such a large software framework is to keep code readable and to not convolute it or create "spaghetti code" (i.e. a big pile of long strings of code). Here, simplicity quite directly translates into code readability and it is important to design code in such a way that it can easily be understood. For example, parameters should be easily and centrally accessible, not forcing the reader to sift through large amounts of code to find out e.g. what the magnitude of any specific scaling parameter or the time step of a solver is.

A good example of code which is hard to simplify is the calculation of gradients and energies in the Hamiltonian. While most methods at some point rely on the energy gradient (and the energy could be calculated from a scalar product of the gradient and the spin), the Monte Carlo method requires the change of energy when a single spin is displaced. At time of writing, this has not been implemented in *Spirit* without the duplication of code for the calculation of energies, as no approach to unify these two concepts without significant performance loss was devised. Especially to abstract away the use of GPU parallelisation in this case is challenging. However, the duplication of the energy calculation functions reduces the legibility of the code and has produced several mistakes which had to be found and fixed.

3.2 HAMILTONIANS

This section deals with a few implementational details of different Hamiltonians. In line with the extrinsic view on the degrees of freedom of the spins and the representation in the $3N$ embedding space (see Section I.2 and Section 2.2), the Hamiltonian classes can implement – if known – the derivatives of the unconstrained (or smoothly extended) Hamiltonians.

The Hamiltonian base class has been equipped with a finite difference approximation (FDA) calculation of the gradient as well as the Hessian. Since the derivatives are only required in the embedding space, they can be trivially implemented. Therefore, quite general Hamiltonians could be implemented without the need to implement the analytical derivatives. Performance may however be an issue for large system sizes, as the finite differences are generally less efficient than an implementation of analytical derivatives.

In *Spirit*, the central difference approximation is used, where

$$G^i_\alpha = \frac{\partial \mathcal{H}}{\partial n^i_\alpha} = \frac{\mathcal{H}(\{\vec{n}\} + h^i_\alpha) - \mathcal{H}(\{\vec{n}\} - h^i_\alpha)}{2h^i_\alpha} \tag{3.1}$$

and

$$\begin{aligned} H_{ij} &= \frac{\partial^2 \mathcal{H}}{\partial \vec{n}_i \partial \vec{n}_j} \\ &= \frac{G^i_\alpha(\{\vec{n}\} + h^j_\beta) - G^i_\alpha(\{\vec{n}\} - h^j_\beta)}{4h^j_\beta} + \frac{G^j_\beta(\{\vec{n}\} + h^i_\alpha) - G^j_\beta(\{\vec{n}\} - h^i_\alpha)}{4h^i_\alpha}, \end{aligned} \tag{3.2}$$

with h^i_α denoting a finite displacement of the spin component n^i_α.

For testing purposes, a Hamiltonian composed of a superposition of Gaussian potentials was implemented, as it allows the creation of arbitrary energy landscapes for a single spin. This can be plotted and used to verify the implementation of a method, as has been done for the GNEB and MMF methods (see Figure 1.7 and Figure 2.4 respectively). Further details on this Hamiltonian are given in Appendix B.

The implementation of dipolar interactions deserves some discussion at this point, as they are the most difficult contribution to the Hamiltonian (I.1) to implement, due to their long-ranged nature (compared to exchange interactions).

The initial implementation in *Spirit* was simply a direct summation with a cut-off radius, an improvement of which was first attempted by implementing the approximation of distant spins inside a macrocell by a single macrospin [52, 178]. While the scaling of the macrocell approach may improve computation time significantly, there is an issue of precision for those spins which are close to the boundary of a macrocell, as a majority of their near-field neighbours are not taken into account with sufficient precision.

Due to these issues, a more robust but scalable method is required. The method of speeding up the calculation of dipolar fields on regular lattices by calculating FFT convolutions is already well-known in micromagnetics [179]. It brings the computational complexity for the evaluation of these interactions from an order $\mathcal{O}(N^2)$ down to an order of $\mathcal{O}(N \log N)$. This was implemented in *Spirit* – under supervision of the author – by Moritz Sallermann as part of his Master Thesis. While non-rectilinear Bravais lattices can be treated trivially, the scheme had to be adapted to cases where a basis cell contains multiple atoms. This is achieved by calculating one FFT per sublattice (i.e. per basis atom) and performing corresponding additional convolutions. High performance, robust FFT libraries are used [180, 181], both on CPU and GPU, which eased implementation and ensures high efficiency.

In order to assert the correctness of the implementations, a set of tests is performed and different evaluations compared against each-other where possible.[3] The stray field of a homogeneously magnetised monolayer is a simple example in which the direct summation can be performed analytically and compared to the numerical results, which should converge against it for increasing system size. Another good test is the reproduction of a calculation from Ref. [182], in which the micromagnetic calculation of the stray field-induced vorticity of a ferromagnetic cube is presented. The results of this calculation using *Spirit* are shown in Figure 3.9.

The squared vorticity is expected to approach the critical field linearly, so that a line can be fitted to extract the precise result from the calculations. The results show the expected behaviour and are already close to the micromagnetic limit with a lattice of 50 × 50 × 50. Increasing the density of lattice sites further leads to a convergence to the expected value for the critical field $h_c = 0.158$.

In ongoing work, the implementation of the fast multipole method (FMM) [183] in *Spirit* is attempted, with the goal of generalising the calculation of dipolar fields to irregular lattices and in order to be able to directly test which method performs the best. Note the Barnes-Hut method, which is a simpler form of the FMM [53].

Note that the comparison of different methods for random configurations with non-symmetric basis cells can already reveal a lot of implementational errors, if present.

[3] For some system sizes direct summation may not be feasible

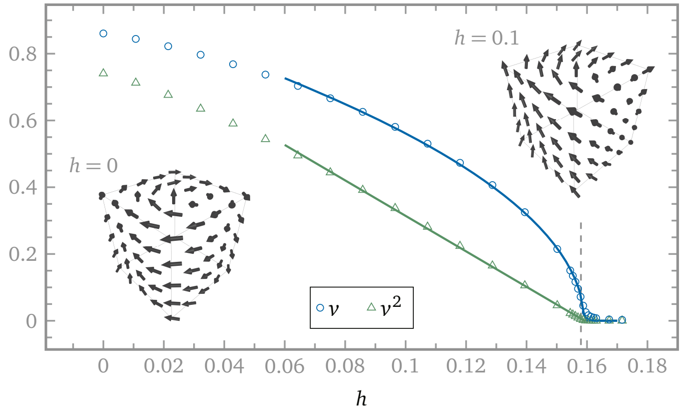


FIGURE 3.9 – Stray field-caused vorticity of a cube

A ferromagnetic simple cubic lattice of size 50 × 50 × 50 with lattice constant $a = 1\,\mathring{A}$ and with dipolar interactions and simple nearest-neighbour exchange of $J = 16.86$ meV. Insets show the cube at $h = 0$ (left) and $h = 0.1$ (right) The vorticity ν (circles) and its square (triangles) are plotted against the reduced external magnetic field h, given in units of $h = B/(\mu_0 M_s)$ where M_s is the saturation magnetisation density. Close to the critical field, the data is fitted (solid lines), giving a value of $h_c = 0.159$. When the lattice density is increased, the micromagnetic limit of $h_c = 0.158$ is approached. Therefore, the results presented in ref. [182] are satisfactorily reproduced. This figure has been published in [47].

3.3 SOLVERS

To simplify the following discussion, we recall the LLG equation (I.34) written in the form of Equation (I.36)

$$\frac{\partial \vec{n}_i(t)}{\partial t} = \vec{n}_i(t) \times \vec{A}_i\left(t, \{\vec{n}_i(t)\}\right), \tag{3.3}$$

where we keep the explicit time-dependence of $\vec{A}$, as the Hamiltonian can be time-dependent, for example when an AC magnetic field is used. This formulation is equivalent to the common notation of ordinary differential equations in the literature

$$\partial_t n(t) = n(t) \times A(t, n(t)) \quad \Longleftrightarrow \quad \partial_t y(t) = f(y, t), \tag{3.4}$$

i.e. $f \,\widehat{=}\, \vec{n} \times \vec{A}$.

3.3.1 *Heun*

Heun's method [184] is a common and illustrative way to solve ordinary differential equations (ODEs) by first calculating an intermediate prediction step and then "averaging" to obtain the final approximation. It can be interpreted as a two-stage Runge-Kutta method [185, 186]. Denoting the time step δt, for an ODE of the form

$$\frac{\partial y(t)}{\partial t} = f(t, y(t)), \quad y(t_0) = y_0, \tag{3.5}$$

the predicted value y^p is first calculated as

$$y^p(t + \delta t) = y(t) + \delta t f(t, y(t)) \tag{3.6}$$

and then the approximation for the next step as

$$y(t + \delta t) = \; y(t) + \; \delta t \frac{f(t, y(t)) + f(t + \delta t, y^p(t + \delta t))}{2}, \tag{3.7}$$

When applied to the LLG equation, where $f \,\widehat{=}\, \vec{n} \times \vec{A}$, this integration scheme obviously does not intrinsically preserve the spin length, requiring the re-normalization of the vectors $\vec{n}_i$ after a given number of iterations, depending on the required precision. Note that Heun's method falls into the category of Runge-Kutta methods, which function analogously and therefore all have this property.

3.3.2 *Depondt*

In order to improve on Heun's method, Ref. [120] proposes to make use of the fact that the spins are only allowed to rotate, by writing an appropriate rotation matrix R, which is calculated directly from the field $\vec{A}$. Applied to Heun's method, the prediction step (3.6) reads

$$\vec{n}_i^p(t+\delta t) = R_i\left(\vec{A}_i(t,\{\vec{n}_i(t)\})\right)\,\vec{n}_i(t)\,. \tag{3.8}$$

To perform the correction step (3.7), one needs the correction field $\vec{A}^c$, which is calculated from the average of the initial and predicted fields,

$$\vec{A}_i^c = \frac{\vec{A}_i(t,\{\vec{n}_i(t)\}) + \vec{A}_i^p(t+\delta t,\{\vec{n}_i^p(t+\delta t)\})}{2}\,. \tag{3.9}$$

From this, in turn, the rotation matrix for the correction step $R_i^c\left(\vec{A}_i^c\right)$ is obtained and the final step of the scheme reads

$$\vec{n}_i(t+\delta t) = R_i^c\left(\vec{A}_i^c\right)\,\vec{n}_i(t)\,. \tag{3.10}$$

Higher order Runge-Kutta schemes could apply this approach analogously.

3.3.3 4th *order Runge-Kutta*

The most well-known of the Runge-Kutta schemes [185, 186] is the classical 4th order Runge-Kutta method, sometimes referred to as "RK4". As always, we start from the initial value problem

$$\frac{\partial y(t)}{\partial t} = f(t,y(t)), \quad y(t_0) = y_0\,. \tag{3.11}$$

The position at the next step is calculated as the weighted average of several increments

$$y(t+\delta t) = y(t) + \frac{1}{6}\left(k_1 + 2k_2 + 2k_3 + k_4\right), \tag{3.12}$$

where the increments are defined as

$$\begin{aligned}
k_1 &= \delta t f(t,y(t)) \\
k_2 &= \delta t f(t+\delta t/2, y(t)+k_1/2) \\
k_3 &= \delta t f(t+\delta t/2, y(t)+k_2/2) \\
k_4 &= \delta t f(t+\delta t, y(t)+k_3)\,.
\end{aligned} \tag{3.13}$$

3.3.4 *Semi-implicit midpoint*

Ref. [118] takes a different approach from the Runge-Kutta class of methods by using an implicit midpoint (IMP) structure to preserve the spin length. The implicit midpoint method, solves differential equations of the form $y'(t) = f(t, y(t))$, $y(t_0) = y_0$ (see Equation (I.36)) and an iteration step is defined as

$$y(t+\delta t) = y(t) + \delta t f\left(t + \frac{\delta t}{2}, \frac{y(t) + y(t+\delta t)}{2}\right). \tag{3.14}$$

For the LLG equation (I.34) and a time step δt this leads us to

$$\begin{aligned} \vec{n}_i(t+\delta t) = \vec{n}_i(t) + \delta t \frac{\vec{n}_i(t) + \vec{n}_i(t+\delta t)}{2} \times \\ \vec{A}_i\left(t + \frac{\delta t}{2}, \left\{\frac{\vec{n}_i(t) + \vec{n}_i(t+\delta t)}{2}\right\}\right). \end{aligned} \tag{3.15}$$

The semi-implicit scheme B (SIB) [118] uses a predictor $\vec{n}_i^p$ to reduce the implicitness of the equation above by replacing $\vec{n}_i(t+\delta t)$ in the argument of $\vec{A}_i$. To preserve spin length the predictor is obtained with the IMP structure.

$$\vec{n}_i^p(t+\delta t) = \vec{n}_i(t) + \delta t \frac{\vec{n}_i(t) + \vec{n}_i^p(t+\delta t)}{2} \times \vec{A}(t, \{\vec{n}_i(t)\}). \tag{3.16}$$

Equation (3.16) can be rewritten as

$$M \vec{n}^p(t+\delta t) = M^T \cdot \vec{n}(t), \tag{3.17}$$

with the matrix

$$M = I + \text{skew}(\vec{A}) = \begin{pmatrix} 1 & -A_z & A_y \\ A_z & 1 & -A_x \\ -A_y & A_x & 1 \end{pmatrix}. \tag{3.18}$$

The right hand side of Equation (3.17) can be easily calculated as

$$M^T \vec{n}_i = \vec{n}_i + \vec{n}_i \times \vec{A}_i =: \vec{a}\,. \tag{3.19}$$

To solve Equation (3.17) we use Cramer's rule. The components $n_{i,\alpha}^p$ of the predicted spin vector $\vec{n}_i^p$ are calculated with

$$n_{i,\alpha}^p = \frac{\det(M^\alpha)}{\det(M)}, \tag{3.20}$$

where M^{α} is the same matrix as M but column α is replaced with the vector $\vec{a}$, for example

$$M^{x} = \begin{pmatrix} a_x & -A_z & A_y \\ a_y & 1 & -A_x \\ a_z & A_x & 1 \end{pmatrix}. \tag{3.21}$$

We now use the predictor $\vec{n}_i^p$ in the IMP step (3.15) to calculate the updated spin vector as

$$\begin{aligned} \vec{n}_i(t+\delta t) = \vec{n}_i(t) + \delta t \frac{\vec{n}_i(t)+\vec{n}_i(t+\delta t)}{2} \times \\ \vec{A}_i\left(t+\frac{\delta t}{2}, \left\{\frac{\vec{n}_i(t)+\vec{n}_i^p(t+\delta t)}{2}\right\}\right). \end{aligned} \tag{3.22}$$

The correction step is analogous to the prediction step (compare eqs. (3.16) and (3.22)), meaning that the scheme (3.20) can be applied to obtain $\vec{n}_i(t+\delta t)$, too.

3.3.5 *Velocity projection*

This description is derived from Ref. [56]. Verlet-like methods [187, 188] generally find application in solving second order differential equations of the form $\ddot{x}(t) = F(t, x(t))$, $x(t_0) = x_0$, $\dot{x}(t_0) = v_0$, such as Newtons equation of motion. In the formulation of this method, implemented in *Spirit*, both the position and the velocity are incremented at each time step:

$$x(t+\delta t) = x(t) + \delta t\, v(t) + \frac{1}{2m}\delta t^2 F(t), \tag{3.23}$$

$$v(t+\delta t) = v(t) + \frac{1}{2m}\delta t(F(t)+F(t+\delta t)). \tag{3.24}$$

The velocity projection is used to accelerate convergence towards local minima, as well as avoid overstepping due to momentum. The velocity at each time step is damped by projecting it onto the force:

$$v \rightarrow \begin{cases} (v\cdot F)F/|F|^2, & (v\cdot F) > 0 \\ 0 & \text{else} \end{cases}. \tag{3.25}$$

Note that the dot product and norm in this equation denote those of $3N$-dimensional vectors.

Using the velocity projection solver, we are no longer solving the LLG equation, but instead a similar equation which converges the system towards local minima (see Section I.4)

$$\frac{\partial^2 \vec{n}}{\partial t^2} = \vec{n} \times \left(\vec{n} \times \frac{\partial \mathcal{H}}{\partial \vec{n}_i}\right). \tag{3.26}$$

The force is simply defined as

$$\vec{F}_i = -\frac{\partial \mathcal{H}}{\partial \vec{n}_i} . \tag{3.27}$$

As the method does not conserve the length of the spins, they should be renormalized after each iteration

$$\vec{n}_i(t+\delta t) \rightarrow \frac{\vec{n}_i(t+\delta t)}{|\vec{n}_i(t+\delta t)|} . \tag{3.28}$$

3.3.6 *Nonlinear conjugate gradients*

The nonlinear conjugate gradients (NCG) method is a sophisticated tool for minimisation problems and can be formulated for nonlinear problems [189], such as the minimisation of the energy of an atomistic spin system, given by a nonlinear Hamiltonian $\mathcal{H}$ (for example Equation (I.1)). This method, like the velocity projection solver, is not suitable for the solution of the dynamical LLG equation and has to be adapted to respect the constraint of constant spin length.

In the language of this method, we define the current position of the system in the configuration space as the point $x = \{\vec{n}_i\} \in \mathcal{M}^{\text{phys}}$ (see also Section I.2 and Chapter 2) and the residual

$$r = -\nabla \mathcal{H} , \tag{3.29}$$

which is simply the effective field. The step size α is a scalar value that minimises

$$\mathcal{H}(x + \alpha d) \tag{3.30}$$

or, equivalently,

$$(\nabla \mathcal{H}(x + \alpha d))\, d = 0 . \tag{3.31}$$

Starting with $d = 0$ and $\beta = 0$, in each iteration of the algorithm, d is updated with the following rule:

$$d_{(i+1)} = r_{(i+1)} + \beta_{(i+1)} d_{(i)} . \tag{3.32}$$

With the step size α, the position x can be updated by

$$x_{(i+1)} = x_{(i)} + \alpha_{(i)} d_{(i)} , \tag{3.33}$$

which in our case requires subsequent re-normalisation of the spin vectors:

$$\vec{n}_i \rightarrow \frac{\vec{n}_i}{|\vec{n}_i|} , \tag{3.34}$$

$$r_{(i+1)} = -\nabla \mathcal{H}(x_{(i+1)}) . \tag{3.35}$$

Since the Polak-Ribière equation for β does not immediately ensure convergence, it is slightly modified:

$$\beta_{(i+1)} = \max\left\{ \frac{r_{(i+1)}(r_{(i+1)} - r_{(i)})}{r_{(i)}r_{(i)}},\ 0 \right\}, \tag{3.36}$$

which will restart the CG if β becomes negative.

In order to perform an exact line search and find the correct value of α in this non-quadratic case, several steps may need to be taken along the line until the minimum is found.One NCG iteration may hence include many line search iterations. For more information on the NCG method in general and line searches in particular, when constrained to a Riemannian manifold, as well as connected problems and solvers see also [190]. This problem was not fully covered in the time span of this work, i.e. the NCG method has not yet been fully implemented in *Spirit*. However, the method has been found to be highly effective [136], which motivates the completion of its implementation for energy minimisations and minimum energy path (MEP) calculations in the future. Notably, formulations in the frame of the Cartesian coordinate representation used throughout this thesis have been made [191–193] and should be taken advantage of.

3.3.7 *Comparison of dynamics solvers*

In this section only dynamical solvers, i.e. not those for energy minimisation, will be compared. In order to judge the precision of the different solvers implemented in *Spirit*, the total error, which is the euclidean norm of the distance between the expected and the calculated spin vector, is plotted over the time step Δt in Figure 3.10.

Recall the analytical solution Equation (I.38) for the angle of the spin precessing in the plane orthogonal to an applied magnetic field

$$\varphi(t) = \frac{\gamma}{(1+\alpha^2)\mu}|\vec{B}|t\,. \tag{3.37}$$

The range of time steps between 10^{-4} ps and 1 ps corresponds to a rotation of the spin by $\mu_\mathrm{s}|\vec{B}|\gamma\Delta t$ from $0,003\,\mathrm{deg}$ to $30\,\mathrm{deg}$ per time step. It is obvious that the SIB and Heun methods have exactly the same precision in this test case with the applied error measurement. The Depondt and 4th order Runge-Kutta solvers have very similar precision, but the Depondt solver appears to match this specific case exactly and it therefore remains precise, even at a time step of 1ps. This is most likely due to the fact that its iterations consist of the application of rotation matrices (see Section 3.3). The numerical precision limit stems from the tradeoff between the additive nature of the error, increasing with the number of time steps, and the decreased error per time step when the time step Δt is reduced.

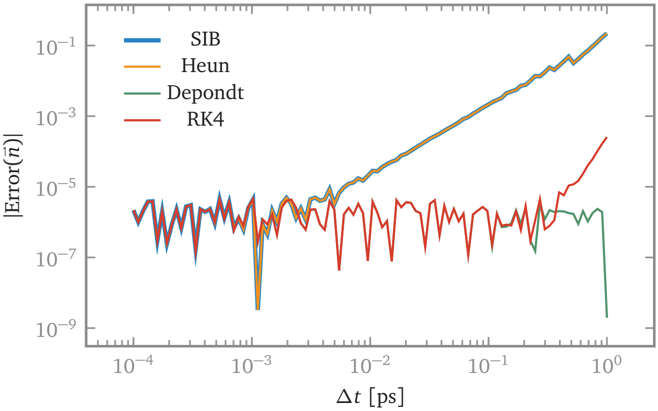


FIGURE 3.10 – Solvers' precession error over time step

The maximum total error of the spin vector $\vec{n}$ over time step Δt is shown for the same relaxation procedure as in Figure I.11 (i.e. the same time interval of 500 ps). Note that the maximum error of the relaxation process is only one possible measure of the precision of a solver. The error has an interesting dependence, as the Depondt solver has its lowest error at a time step of 1 ps, while the other solvers show the expected behaviour of a decreasing error for smaller time steps until the limit of numerical precision is reached. This is likely due to the nature of the rotation matrix in the Depondt solver (see Section 3.3) and the pure in-plane rotation in this test case (compare Figure 3.11). Note also that the SIB and Heun solvers coincide everywhere.

Obviously the precession-only test case is educative, but does not provide a good measure of precision of the solvers. In Figure 3.11, we show again the dependence of the total error on the time step Δt. Recall the analytical solution Equation (I.39):

$$\begin{aligned} n_z(t) &= \tanh\left(\frac{\alpha\gamma}{(1+\alpha^2)\mu}|\vec{B}|t\right) \\ \varphi(t) &= \frac{\gamma}{(1+\alpha^2)\mu}|\vec{B}|t \\ n_x(t) &= \cos(\varphi(t))\sqrt{1-n_z^2(t)} \\ n_y(t) &= \sin(\varphi(t))\sqrt{1-n_z^2(t)}. \end{aligned} \tag{3.38}$$

The total error of the damped precession over the time step Δt shows a dependence similar to that of the undamped precession. The fact that the semi-implicit solver has the highest error in this test is surprising and requires further investigation.

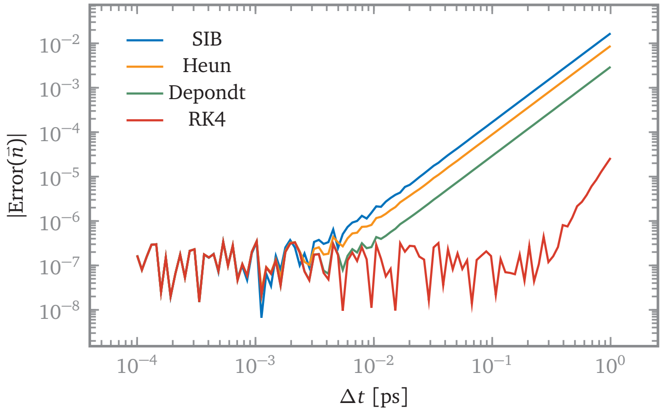


FIGURE 3.11 – Solvers' error dependence on time step

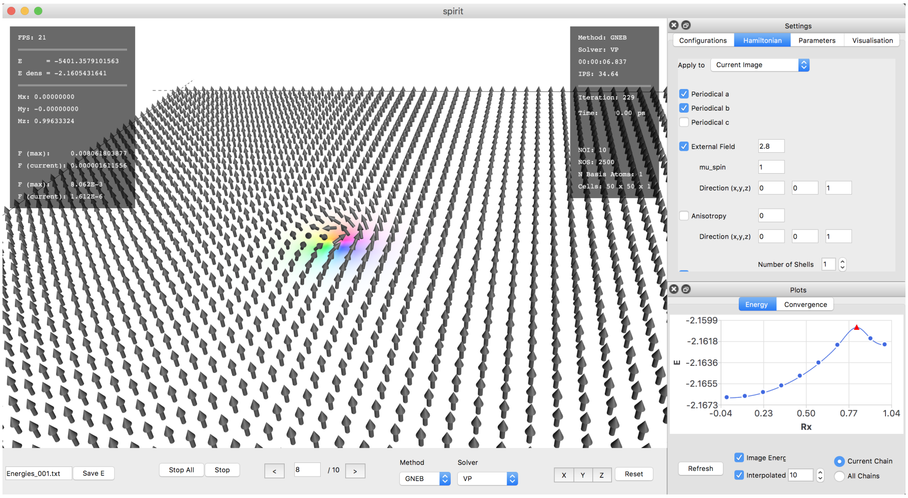
The maximum errors of the x- and z-components respectively are shown for the same relaxation procedure as in Fig. I.12. In this measure, the 4th order Runge-Kutta solver obviously shows the lowest error above the limit of numerical noise, which it reaches already at time steps an order of magnitude larger than for the other solvers. Note that the maximum error of the relaxation process is only one possible measure of the precision of a solver.

In order to make sure that no change in the implementation compromises the correctness of any of the solvers, the shown precession tests are reproduced as unit tests in the continuous integration. To also test the minimisers and the convergence of a complex spin texture, a skyrmion is relaxed with each solver and the energy compared against the numerically known value.

3.4 GRAPHICAL USER INTERFACE

The graphical user interface (GUI) of *Spirit* is the core productivity tool of the framework. Its features are intended to enable quick testing and execution of the various implemented parameters and methods and to combine them into complex calculations. Note that in addition to describing the implemented features, this section is also meant as a general guide, giving perspective beyond what has been implemented in *Spirit* so far.

Use cases for such a simulation code can vary greatly and it is thus important to keep a certain level of abstraction and generality. In the end, the amount of methods and parameters should not overwhelm the user, but appear naturally and be intuitively accessible.



```
FPS: 21

E      = -5401.3579101563
E dens = -2.1605431641

Mx: 0.00000000
My: -0.00000000
Mz: 0.99633324

F (max):      0.008061803877
F (current): 0.000001611556

F (max):      8.062E-3
F (current): 1.612E-6
```

```
Method: GNEB
Solver: VP
00:00:06.837
IPS: 34.64

Iteration: 229
Time:     0.00  ps

NOI: 10
NOS: 2500
N Basis Atoms: 1
Cells: 50 x 50 x 1
```

Apply to Current Image

☑ Periodical a
☑ Periodical b
□ Periodical c

☑ External Field 2.8
mu_spin 1
Direction (x,y,z) 0 0 1

□ Anisotropy 0
Direction (x,y,z) 0 0 1

Number of Shells 1

Energies_001.txt

8 / 10

Method GNEB

Solver VP

☑ Image Energ
☑ Interpolated 10

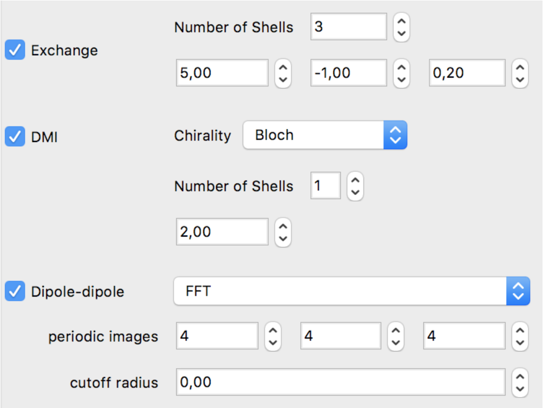
● Current Chain
○ All Chains

FIGURE 3.12 – Screenshot of Spirit GUI running the GNEB method

In addition to the visualisation, information panels show useful data, such as the energy of the current image, the maximum force currently acting along the chain or the number of iterations per second. Controls to change the simulation parameters, the Hamiltonian or the visualisation and to set initial states are given on the right. A graph shows the energies of images along the chain, the selected being colored red (indicated as a triangle because it is set as climbing image). The control bar in the bottom allows for switching between images on the chain and selecting the method to be applied and the solver to be used.

A major challenge in this endeavour is the presentation of information on a single spin system as well as a chain of systems at the same time. A significant advantage of handling a complete chain is the ability to run multiple single-system iterations in parallel and monitoring them at the same time. These issues were resolved by presenting one system of the chain in the visualisation, while also showing a plot of the energies of the chain with an indication of the currently selected system, as shown in Figure 3.12. Together with an intuitive copy-paste feature, enabling the simple creation, reordering and manipulation of chains of systems, this makes all calculations easy to monitor in the same GUI.

In the context of parallel calculations and different methods potentially conflicting with each-other, one has to take care to adequately protect the user from himself, in the sense that at any time, the available user input should be valid. For example, one should not be able to start LLG calculations on a chain, where a GNEB calculation is already running. Also, certain input needs time to be processed – for example the calculation of eigenmodes – and the GUI should deactivate the appropriate controls for the time it takes to finish the operation and block further operations, such as the removal of the corresponding image from the chain. Due to the increasing complexity with every added method and parameter set, this can become complicated to keep in check.



☑ Exchange
Number of Shells 3
5,00 -1,00 0,20

☑ DMI
Chirality Bloch
Number of Shells 1
2,00

☑ Dipole-dipole FFT
periodic images 4 4 4
cutoff radius 0,00

FIGURE 3.13 – Interactions input in the Spirit GUI

Screenshot of the GUI of *Spirit*, showing the input of three neighbour shells of exchange, one neighbour shell of DMI and the usage of FFT convolutions with 4 periodical images in each direction for the calculation of the dipolar interactions. Note that the cutoff radius is only used if the direct summation is selected as calculation method for dipole-dipole interactions.

The input of interactions from the GUI might seem trivial, but the fact that there are many different ways to represent the same input means that this feature should not be too limited. Due to the added complexity of higher order interactions, only pairwise parameters can be set, such as neighbour shells of exchange and DMI, as well as the method and parameters for the dipolar interactions. This is shown in Figure 3.13. While exchange and DMI can, in general, be specified pairwise in order to provide greater generality, the GUI does not provide this feature, as it would involve additional challenges in the design of the interface.

It is difficult to make such functionality available through multiple different interfaces, namely input files, application programming interface (API) and graphical user interface (GUI). A rule which the author found helpful in the design of these different interfaces was that it should make no difference through which interface a parameter was specified and when – the result should always be the same, also including runtime performance. For example, neighbour shells and explicit pairs should be equally fast.

In order to provide advanced functionality to initialise complex states and intervene in running calculations, an interaction tool was implemented into the GUI of *Spirit*, enabling the dragging or copying and pasting of a cylindrical region of the system. This enables the easy generation of complex shapes or, for example, skyrmion lattices without much effort – potentially saving a lot of time for example when the outcome of various initial conditions is still unclear and has to be explored. This is illustrated in Figure 3.14.

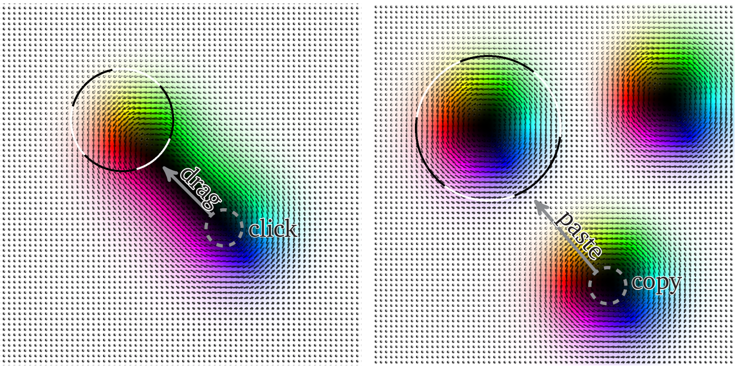


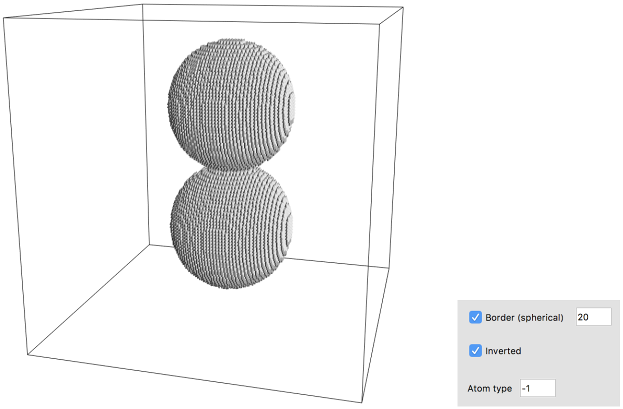
FIGURE 3.14 – Interactively changing spin configurations in the GUI of *Spirit*

An interactive tool of *Spirit* is shown *dragging* a skyrmion (left) and *copying* and *inserting* multiple skyrmions (right). This tool can be especially useful when creating a chain of images representing a specific initial guess for the transition, which would be complicated even to script. Also, GNEB calculations may often show unexpected or undesired behaviour, for example due to the appearance of translational or rotational modes. The transitions of skyrmion rings, see Figure 1.10, are a good example for this. It can often be very helpful to be able to directly intervene and correct bad paths segments quickly after they appear. This may, in turn, reduce the number of required images and therefore the computational time needed.

Since its implementation, this tool has been used throughout this work to generate the initial states for GNEB paths and has been found by the author to be highly useful also to correct running calculations, especially in complex 3D systems as presented in Chapter 4.

Analogous to the selective orientation of spins and generation of patterns, one might want to create a pattern of vacancies or pinned spins.

As shown in fig. 3.15, through combination of different settings and features of *Spirit*, it is quite easy to create complicated geometries and quickly test ideas. This – once again – greatly reduces the users effort needed in setup, post-processing and correction steps. Less time is wasted on trial and error and the intuition for studied systems is again enhanced through the immediate visual feedback.



☑ Border (spherical) 20

☑ Inverted

Atom type -1

FIGURE 3.15 – Creating simple and complex geometries in *Spirit*

A spherical arrangement of spins can be easily cut out of a system (left) by setting all other sites to "atom type -1", corresponding to vacancies without any other effect on the system parameters. This can be done very easily from the *Spirit* GUI by setting the region and atom type (right) and the click of a button.

In addition to the specification of pinned boundary layers and similar initial conditions through an input file, as well as setting them in the GUI, as shown in Figure 3.15, a second interaction tool was implemented into spirit, allowing to pin or change the atom types of spins in a cylindrical region. This is illustrated by the insertion of a large region of vacancies by dragging the interaction tool over the sample, shown in Figure 3.16

There is a challenge in the implementation of such features, which should be mentioned: depending on how the input is framed, the user might expect different things to happen with vacancies or pinned spins when the geometry, atom types or the spin configuration are changed. To achieve consistency, making such a tool intuitive to use, is important. For example, what should happen if the user adds a second atom to the basis cell at runtime – is a vacancy applied to the whole cell or is the new atom initialised with some default atom type? Enabling more and more complex use cases and adequately treating the various interactions between all of them is a challenging task and will require significant improvements over the current GUI of *Spirit*.

Finally, one of the most compelling features of *Spirit* is its powerful visualisation. The visualisation was created by Florian Rhiem in collaboration with the author and written as a standalone C++ library [60]. The integration into the GUI of *Spirit* was performed by the author. The visualisation capabilities of *Spirit* have already found application for multiple publications [46, 47, 58, 61, 194, 195], several of which were not in collaboration with the author of this work. Though the interpolated surface colouring is already useful, for example in the visualisation of skyrmion eigenmodes by visualising its contours, it unfolds its real usefulness in 3D systems, where post-processing becomes increasingly cumbersome. See Figure 2.3 for an example of a

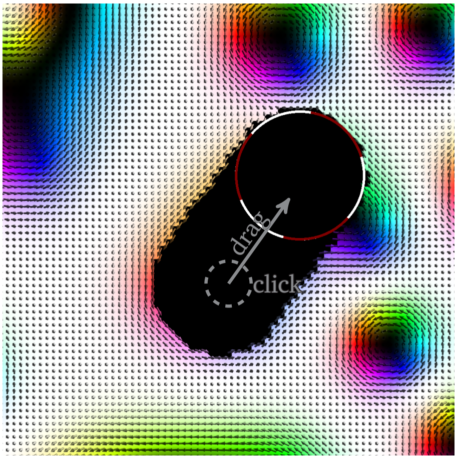


FIGURE 3.16 – Interactively inserting vacancies in the GUI of *Spirit*

An interactive tool of *Spirit* is shown "erasing" spins from a system. Though mostly useful for testing purposes, this tool allows for the easy creation of complex geometries and experimentation on them – without the effort of creating lists of atoms, or even remembering specific input keywords for such features. This feature was used to produce the notch in the racetrack shown in Figure I.15 with just a few clicks.

more complex eigenmode, where the live visualisation becomes highly useful. See also Chapter 4 for more examples of complex visualisations of 3D textures using *Spirit*.

Regarding the quality of the visualisations presented in Figure 3.17, it should be stressed once more that these publication quality visualisations are available in real time during simulations and do not need to be generated by post-processing simulation output.

There are plenty of different techniques one might want to use to visualise spin systems and the potential to combine them increases the complexity of the implementation and GUI even further. It is therefore an important task to make the interface easy to use without restricting features. While the author enabled the use of several features in the same visualisation and the combination of multiple isosurfaces, in hindsight there are many opportunities left open. Ideally, the GUI should provide the ability to add arbitrarily many renderers (a renderer representing one isosurface, visualisation of arrows or any other feature), where the settings of the colour map, data resolution, filters, light and shadows and all other settings should be available both globally and once per renderer. This would enable the full complexity of almost arbitrary visualisations, without revealing the complexity to the user unless requested – i.e. keeping it easy to use.

As was shown in this section, the graphical user interface (GUI) of *Spirit* can greatly

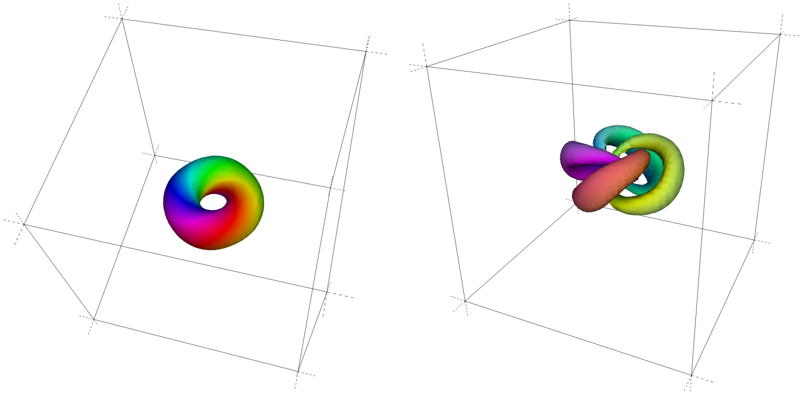


FIGURE 3.17 – Hopfion isosurface visualisations

A Hopfion in a periodic cube is visualised with the isosurface for $n_z = 0$ (left) and the ring-shaped isosurfaces for $n_x = 0.9$, $n_x = -0.9$, $n_y = 0.9$ and $n_y = -0.9$. The coloured rings can be seen to mutually form single links, a way of counting the order of the Hopfion. See [46] for more examples of these isosurfaces, generated using *Spirit*.

increase user productivity in comparison to the currently far more common approach of programmatically setting initial conditions for a calculation and subsequently running them without direct visual feedback. The benefits shown here will surely be adopted by new software being developed in this field and hopefully pre-existing software, too.

As a future endeavour, it would be a good idea to make use of the Python interfaces of both *Spirit* and *VFRendering* and write a new GUI in Python. The ability to rapidly prototype new features and extend functionality would be greatly beneficial. Furthermore, one could compile various different versions of the *Spirit* core library – with and without CPU parallelisation, with GPU backend, with and without support for defects or pinned spins – and load the desired version at runtime and, if desired, switch between them. Finally, it would make the distribution of the GUI version of *Spirit* far easier across all platforms.

Another important addition to the graphical features inside the *Spirit* software framework will be post-processing features. While it was stated repeatedly that various features of the GUI make post-processing obsolete, there are several features which should be added for greater convenience, such as the ability to automatically generate a smooth transition movie from a chain of spin systems by interpolating between the spin configurations and creating screenshots of each configuration to produce a given framerate for a movie. Especially when a calculation was run on a remote machine or cluster, the ability to automatically generate images or movies in *Spirit* could be valuable. Stream-tubes, which have already been implemented in *VFRendering*, but not yet in the GUI of *Spirit*, often find application in fluid mechanics and similar simulations, but can also be used to generate interesting visualisations of spin systems.

3.5 APPLICATION PROGRAMMING INTERFACE

The application programming interface (API), is arguably the most important abstraction in the core library of *Spirit*. It enables the flexibility and versatile use of this code, and provides an important separation of components, namely the core library and the user interfaces, as shown in Figure 3.1. This section will detail some design decisions and the most important aspects and functionalities of the underlying API, written in the C programming language, as well as some lessons learnt. Finally, some aspects specific to the Python language bindings are discussed.

3.5.1 *C API layer*

The essential part of the API is the C layer. As this language is widely supported by other programming languages, it is comparatively easy to use from many other languages, such as Python (see section 3.5.2) or JavaScript – the latter of which was used to create a web-based user interface to *Spirit*, which can even be used on current-day smartphones. This versatility guided the decision for the use of C.

An important consideration for any API is thread-safety, meaning it should be usable from multiple threads at the same time. To this end, an opaque *State* object is used, which all API functions require and access, while using mutex locks to ensure that resizing of data arrays will not lead to segmentation faults (for example when the geometry of a system is changed).

The following example shows how to set an external field, two shells of exchange interactions and one shell of DMI

```c
float dir[3]{0, 0, 1};
Hamiltonian_Set_Field(p_state, 2.0, dir);
float jij[2]{1.0, -0.2};
Hamiltonian_Set_Exchange(p_state, 2, jij);
float dij[1]{0.45};
Hamiltonian_Set_DMI(p_state, 1, dij);
```

LISTING 3.1 –

C API example

3.5.2 *Python API*

Within the scope of this work, many reasons can be stated why to prefer a scripting language, such as Python, when programming complicated calculations. The main differences are

- performance will generally be lower
- overhead of having to re-compile after any change is avoided
- less strict interpretation rules
- simpler syntax
- functional programming features

When the computationally significantly expensive parts are executed in compiled code, these features are typically big time-savers. This means getting the best of both worlds and is achieved by wrapping, or binding, the C API in a layer of Python.

For example, to set an external field, two exchange shells and one DMI shell:

```
hamiltonian.set_field(p_state, 2.0, [0,0,1])
hamiltonian.set_exchange(p_state, 2, [1.0, -0.2])
hamiltonian.set_dmi(p_state, 1, [0.45])
```

LISTING 3.2 –

Python API example

3D MAGNETIC TEXTURES 4

"Heutzutage haben die Wissenschaftler mehr Fantasie
als die Verfasser von Kriminalromanen."

— Werner Heisenberg

As pointed out in the Introduction Section I.1, the field of nanomagnetism is moving towards three-dimensional systems, both in simulations [196] and experiment [197], and towards increasingly complicated device designs. *Spirit* was developed, among other goals, to be a tool for the theoretical exploration and quantitative study of complex 3D magnetic textures. This chapter will show the usage of several features of the software in order to study known states and reveal new ones.

As a typically suitable model for isotropic chiral magnets, the classical Heisenberg model is used throughout this chapter, which is defined by the following Hamiltonian [17, 136]:

$$E = -\mu_s \vec{B}^{\mathrm{ext}} \sum_i \vec{n}_i - J \sum_{\langle ij \rangle} \vec{n}_i \cdot \vec{n}_j - \sum_{\langle ij \rangle} \vec{D}_{ij} \cdot (\vec{n}_i \times \vec{n}_j)\,, \tag{4.1}$$

where $\vec{n}_i = \vec{m}_i/\mu_s$ is the unit vector of the magnetic moment at lattice site i, $\vec{B}$ is an external magnetic field, the $\langle ij \rangle$ denote nearest-neighbour pairs, J is the Heisenberg exchange constant and $\vec{D}_{ij}$ is the Dzyaloshinskii-Moriya vector defined as $\vec{D}_{ij} = D\vec{r}_{ij}$ with the scalar DMI constant, D and the unit vector $\vec{r}_{ij}$ pointing from site i to site j.

In order to keep the generality of the results presented in this chapter, as well as consistency with earlier studies, the size of the simulated domain is always given in reduced units of distances with respect to $L_\mathrm{D} = 2\pi aJ/D$ – the lowest period of an incommensurate spin spiral, where a is the cubic lattice constant – and the external magnetic field is given in reduced units with respect to $B_\mathrm{D} = D^2/(\mu_s J)$ – the critical field at which the bulk system reaches the field polarized ferromagnetic sate. For external magnetic fields in the range $0 < B < B_\mathrm{D}$ the ground state of the bulk crystal corresponds to the conical state – a helical spin spiral with $\vec{k} \parallel \vec{B}$ and magnetization tilted towards the direction of $\vec{B}$.

4.1 THE CHIRAL BOBBER

Quite recently, a new, metastable magnetic state was introduced in the context of 3D nanomagnetism: the chiral bobber [136] – a finite skyrmion tube extending from a Bloch point to a free surface of the sample. This state was recently confirmed in experiments [194]. A visualisation of a chiral bobber in *Spirit* is shown in Figure 4.1.

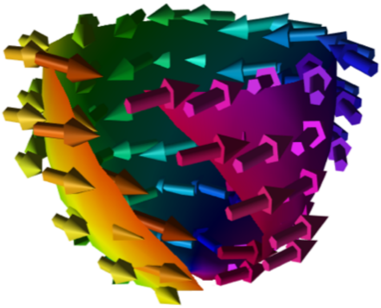


FIGURE 4.1 – Chiral bobber visualised in *Spirit*

A chiral magnetic bobber, visualised in the graphical user interface (GUI) of *Spirit*. An isosurface for $n_z = 0$ is drawn, colors indicating the in-plane orientation of the magnetic moments, which are visualised by arrows. The configuration is cut open to better show its full structure. Note that spins inside are tilted against the z-axis, while they are tilted towards the z-axis on the outside. At the bottom of the isosurface the colors meet in the Bloch point. This shows the complexity of the bobber's structure, which can easily be handled in *Spirit* without any need for post-processing.

Bloch points (BPs) [198] are topologically nontrivial objects, representing point singularities in the magnetisation field [199, 200] and have recently become commonly referred to as quasi-monopoles or monopoles [201, 202]. In its close vicinity, the magnetisation is pointed radially away from the BP, though the magnetic flux through the surface surrounding the point is equal to zero. Note that even in the micromagnetic theory, where the magnetization vector becomes undefined at the BP, the energy does not diverge [200].

The skyrmion tube and chiral bobber are especially interesting states, as their combination was proposed for a new racetrack memory design, where the bits "0" and "1" are encoded by skyrmions and bobbers respectively [194]. This design has a number of advantages over current skyrmion racetrack designs, the most important of which are the fact that skyrmion and bobber can be distinguished by measurement, making the data readable, and that there is far less of an issue with spacing, as no data is represented by free space. Especially when pinning, for example due to defects, starts to play a role, it becomes important that there is no requirement for all the data to move at the same speed all the time. Recently, theory of chiral orbital magnetism was employed to try and enable the unambiguous differentiation between skyrmions and bobbers by electrical measurement [196]. It is shown that the transport properties of this state have unique qualities and can be tuned by external fields, for which the topology of the Bloch point and the object itself play a major role.

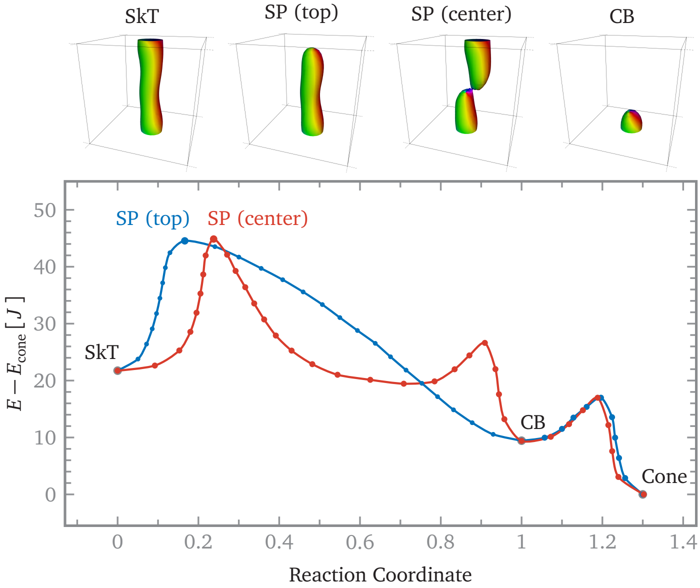


FIGURE 4.2 – GNEB path for skyrmion collapse with intermediate bobber

A 30 × 30 × 30 simple cubic lattice with periodical boundaries in x and y directions and free surfaces in z-direction at an external field of $B = 0.8\ B_{\mathrm{D}}$. The minimum energy paths (MEPs) for the collapse of a skyrmion tube through nucleation of a Bloch point at the surface (blue curve) and a pair of Bloch points at the center (red curve) are shown. The skyrmion tube is either disconnected from one of the surfaces through radial shrinking in the nearby layers, leading to the nucleation of a Bloch point, or it is cut in half by the nucleation of a pair of Bloch points in the center. In the latter case a pair of chiral bobbers appears along the MEP, as well as a saddle point for the collapse of one of them, and in both cases there is a single chiral bobber and its collapse saddle point on the path. Note that the slight differences in the collapse of the bobber between the two paths come from different initial paths. This figure has been published in [47].

Spirit not only has the ability to visualise these states in real time, but to apply both LLG dynamics simulations of such a racetrack with electric currents as well as the GNEB method to find energy barriers for the various possible transitions in such a system. While LLG simulations were performed, no conclusive results could be obtained within the time frame of this work and this section will deal only with GNEB calculations. The parameter set used throughout this section is in accordance with a calculation presented in ref. [136], for which a minimum energy path (MEP) was presented. For these calculations the specific parameters are $J = 1$ and $D = 0.45\ J$, meaning that the incommensurate spin spiral has a period of $L_\mathrm{D} = 13.96\ a$. The first calculation, presented in Figure 4.2, is an example and a proof of concept for the GNEB method in 3D.

While Ref. [136] reports the process of nucleating a pair of Bloch points to cut the skyrmion tube in half and produce a pair of bobbers, Figure 4.2 shows that the nucleation of a single Bloch point at a free surface of the thin film is also a possible MEP. At the given field, both processes have almost equal energy barriers of $\Delta E_\mathrm{center} = 23.13\ J$ and $\Delta E_\mathrm{surface} = 22.81\ J$. The collapse of a chiral bobber has an energy barrier of $\Delta E_\mathrm{bobber} = 7.55\ J$.

In order to obtain the exact energy barriers, the climbing image method [203] was used (see also Section 1.2) and – analogous to what is suggested in the reference – the spring force modulated for an even distribution of images along the curve $E(Rx)$. The latter improves the convergence of the saddle point images quite significantly, as the finite difference tangents become more accurate. The fact that the climbing image method should be used when calculating energy barriers, especially when the resolution close to the maximum is low, is illustrated by the fact that these results give a ratio of the energy barriers between the collapse of the bobber and the Bloch point nucleation of only 3.3, while ref. [136] – not using climbing images – reports a ratio of 4.3.

It should also be noted that these calculations are made more difficult by the presence of a conical phase background – corresponding to the ground state of the system – which introduces additional modes with little or no energy cost associated to them. Such zero or very low energy modes can significantly slow down the convergence to the minimum energy path.

The following calculations were made in order to further analyse the nucleation of Bloch points, but also to determine a set of quantitative reference results for this nontrivial kind of system, against which other codes might be tested. The dependence of the various energy barriers on the external magnetic field was calculated and the results are shown in Figure 4.3.

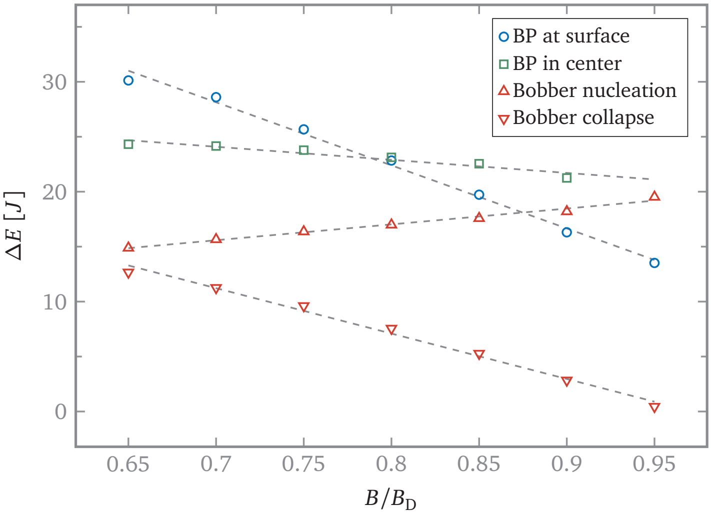


FIGURE 4.3 – MEP barriers of the skyrmion tube and the bobber

Energy barriers of the two MEPs shown in Figure 4.2 (cube of size $(30\times30\times30)$, periodic boundary conditions in the xy-plane) are plotted against the applied magnetic field B. The nucleation of a BP in a skyrmion tube at the surface of the thin film (blue circles) and in the center of the skyrmion (green squares) are shown, intersecting approximately at $B = 0.8B_\mathrm{D}$. The nucleation of an isolated bobber at a surface of the thin film (red upwards triangle) and its collapse (red downwards triangle) are shown as well. This figure has been published in [47].

The results reveal a crossover between the two Bloch point nucleation mechanisms in a skyrmion tube: for increasing external magnetic field it becomes favourable to nucleate just one Bloch point at a free surface of the thin film, while at lower fields, where the out of plane component of the conical background is decreased, it becomes more favourable to nucleate a pair in the center of the skyrmion tube. The energy barrier for the collapse of the bobber goes to zero approximately at the critical field B_D, meaning that the bobber, at least in this model, can only be stabilised inside the conical phase. Finally, in contrast to a radial collapse of the skyrmion tube, the nucleation of a Bloch point at a free surface, producing a bobber, also becomes easier towards lower external field.

4.2 IN SEARCH OF NEW LOCALISED STATES

The interesting properties of BPs, mentioned in the previous section, warrant further investigation. In the course of this work, the formation of structures containing BPs was investigated further and a new 3D-localised particle-like state, composed of two Bloch points coupled together with the center resembling a skyrmion, was discovered.

Due to the usually elliptical isosurface of these objects and the Bloch points at the poles of the surface, as shown in Figure 4.4, they will in this work be referred to as magnetic globules.

Results presented in this chapter have been published by the author in Ref. [61].[1]

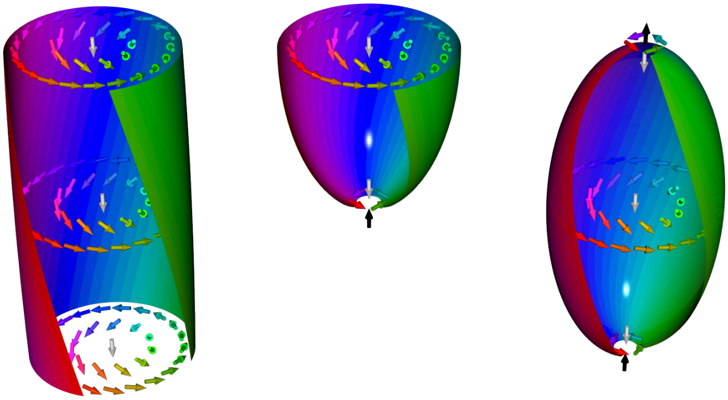


FIGURE 4.4 – Comparison of skyrmion bobber and globule

The general structures of the vectorfields of the skyrmion tube (left), the chiral bobber (middle) and the magnetic globule (right) are schematically visualised. Image provided by Nikolai Kiselev.

This spin configuration represents a novel phenomenon, as the stability of these objects does not depend on geometric confinement. This point will be proven in the following. As shown in Figure 4.5a), it is possible for globules to attach themselves to a skyrmion tube – also several at a time, at different positions and with varying distances between the Bloch points. By creating a special set of boundary conditions, a bulk state containing a defect line of the conical phase can be calculated.[2] In x-direction, the boundaries are set to represent the analytical solution of the conical phase, where on one side there is one repetition less. In y-direction, the boundary conditions can simply be periodical. In z-direction, however, spins at the boundary have mirrored interactions, pointing back into the sample, creating the effect of an infinite continuation of the same state, without periodical repetition of the defect line. Special care has to be taken in order to obtain a smooth transition to this boundary condition, in order to avoid numerical artefacts. As illustrated by Figure 4.5 b) and c), the globule can also attach to such a defect line and exist effectively as an isolated state.

Figure 4.5a) shows an isolated skyrmion tube within the conical phase (which is evident by the spiral-formed distortion of the isosurfaces for $\vec{n} \parallel \vec{B}$ and $\vec{n} \perp \vec{H}$) with several globules of varying lenght attached to it, while b) and c) show a defect line

[1] The paper is on arxiv and is under review at Physical Review Letters

[2] The "bulk" boundary conditions were implemented by Filipp Rybakov in his code and the corresponding calculations therefore not performed with *Spirit*

of the bulk conical phase with a single globule attached to it. It was previously believed that such a pair of Bloch points could not form a stable, coupled configuration without the presence of geometric confinement [204, 205]. This is improved upon by these findings that the globule can exist as a metastable state, when located at such inhomogeneities. When isolated and coupled to a defect line, the globule is found to be metastable between $0.5H_\mathrm{D}$ and $0.85H_\mathrm{D}$.

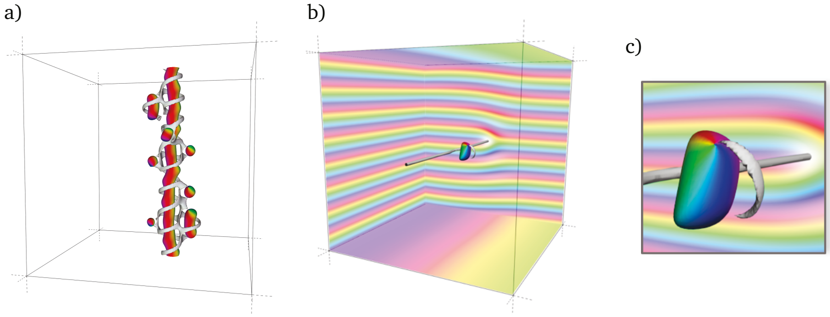


FIGURE 4.5 – Magnetic globules in bulk systems

a) Several globules coupled to a single skyrmion tube in a bulk system. The isosurfaces show where the vector field is pointed in-plane (coloured) and close to z-direction (white). b) Isolated globule stabilised within the conical phase at an edge dislocation. c) a closer view of (b). These images have been published in [61].

As the previously mentioned bulk boundary conditions have not been implemented in *Spirit*, the GNEB calculation of an energy barrier for the collapse of this structure was not possible. Instead, the stability of a globule coupled to a skyrmion tube was studied further. The computational cost of the GNEB method prevented using large systems, but the size was chosen to be sufficient for conclusions to be drawn from them. Figure 4.6 shows the results of a minimum energy path for the nucleation of a pair of Bloch points, which couples to a close-by skyrmion tube, and the subsequent increase of distance between the Bloch points.

The results clearly show local energy minima for certain periods of the distance d_BP between the Bloch points and energy barriers for the collapse of these states. This means that the globule attached to a skyrmion tube is indeed a metastable state, which cannot be destroyed by arbitrarily small excitations. From the images Figure 4.6 b) - d), it is obvious that d_BP is quantised with a period of $\sim L_\mathrm{D}$, while the MEP shows that the lowest energy state always corresponds to the state with the smallest $d_\mathrm{BP} \lesssim L_\mathrm{D}$ (which can slightly vary with the applied magnetic field). The energy increases linearly with d_BP and tends towards the energy of two infinitely long, coupled skyrmion tubes. Such coupled skyrmions have recently been shown to appear due to the attractive interaction between skyrmion tubes at $B < B_\mathrm{D}$ [195, 206, 207] and it is conjectured that the same effect is responsible for the stability of the globule in this case. As there is an energy barrier for the collapse of the globule, corresponding to an annihilation of the pair of Bloch points, a single skyrmion tube can host several globules, as was already shown in Figure 4.5a).

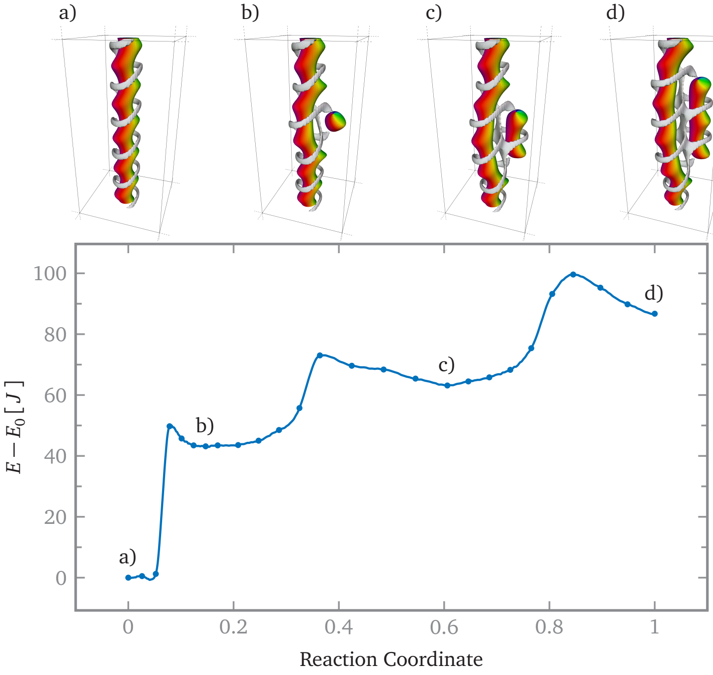


FIGURE 4.6 – Skyrmion-globule GNEB path in the bulk

Globules coupled to a skyrmion tube in a periodical bulk system, calculated at $B_{\text{ext}} = 0.45 B_{\text{D}}$. A coupled pair of Bloch points (BPs) may attach to a skyrmion tube in a bulk system, allowing it to be metastable at varying distances between the BPs. The minimum energy path (MEP) for the nucleation of a pair of BPs next to the skyrmion and the subsequent increase of distance between them, i.e. the elongation of the glouble, is plotted. The energy increases linearly with the distance between the Bloch points, converging to the energy of two coupled skyrmion tubes in the bulk. Finite energy barriers are present for both the increase and the decrease of distance between the BPs. This figure has been published in [61].

A natural conclusion from the above results is that inhomogeneities could stabilize globules in general and a logical next step is to seek out systems where such modulations could be predictably stabilised. Instead of relying on metastable magnetic textures to be present in the sample, they can be naturally provided by the presence of so-called edge modulations, which are always present in finite size samples [13, 208–211]. First, it is important to check once more that it is not geometrical confinement, which stabilises the globule.

As a system to examine, a nanowire with periodical boundary conditions along its length was chosen – for simplicity the wire has a square shape. Figure 4.7 shows the stability range of the globule over the wire width W for a set of external magnetic fields B. It should be noted that for energy minimisation of large system sizes of several millions of spins, the CUDA code of Filipp Rybakov was used due to its superior performance and implementation of a nonlinear conjugate gradients (NCG) solver (see also Section 3.3).

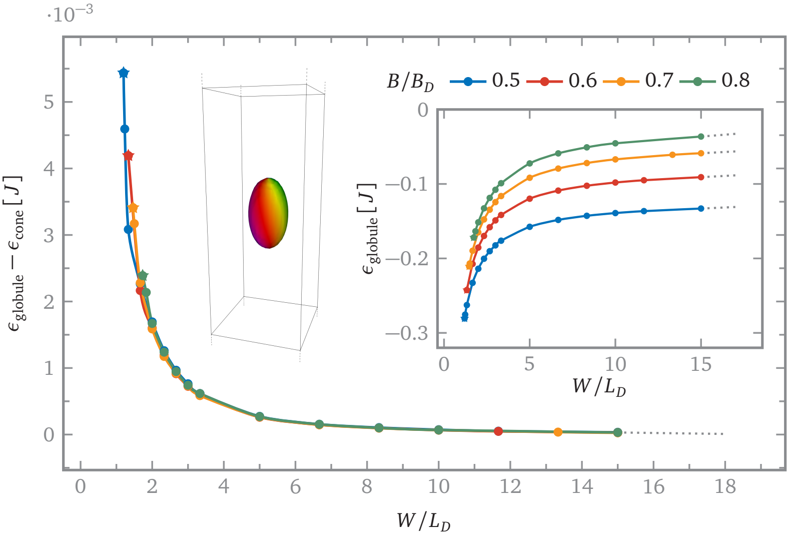


FIGURE 4.7 – Energy of the globule in an infinite nanowire

A nanowire of varying width W at a set of applied fields B, where $L_z = 10L_\text{D}$ and W up to $15L_\text{D}$ with periodical boundary conditions along the z-axis, see left inset. The dependence of the difference between the energy density of the conical ground state and the globule is shown. The right inset shows the same dependence in absolute units of energy. Stars on the left ends of lines denote the collapse of the globule, indicating that it may remain stable towards infinite width of the wire, as it can couple to a side or a corner of the wire (see Figure 4.8). The convergence of the difference in energy density means that the globule is indeed a well-localised state and does not fill the entire available space.

Figure 4.7 also shows the convergence of the energy density of the globule in the nanowire for increasing width W. This clearly indicates that, despite the presence of two magnetic singularities, the globule represents a well-localised state, which does not extend to fill the entire available space. In contrast to actual geometrical confinement, which would restrict the size of the globule, edge modulations allow for truly 3D-localised globules.

Since the globule is – in this scenario – coupled to edge modulations, its equilibrium position is near the edge of the nanowire, as shown in Figure 4.8 for a very wide nanowire. As is shown, for a square nanowire where $W \gg 2L_\mathrm{D}$, the globule is attracted to the corner.

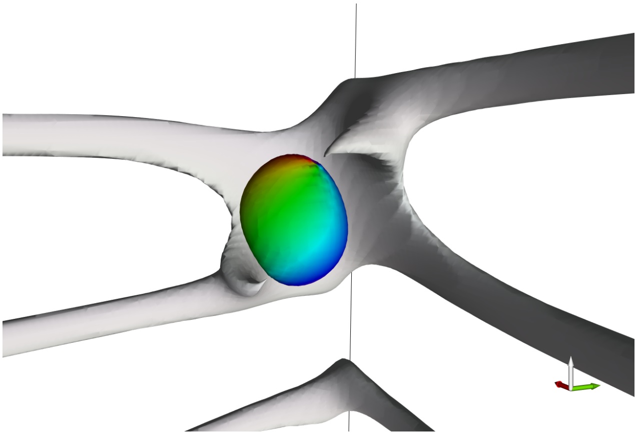


FIGURE 4.8 – Globule attached to nanowire edge state

Magnetic globule coupled to edge modulations in a wide nanowire with $W = 3L_\mathrm{D}$ and $B = 0.6B_\mathrm{D}$ (see Figure 4.9). Isosurfaces for $n_z = 0$ (coloured) and $n_z \approx 1$ (grey) of the globule and edge state are shown for a nanowire of large width. The globule is always trapped between two branches of isosurfaces corresponding to spins pointing along the external field.

The next question is in which range of parameters such states might be found. To answer this, the diagram in Figure 4.9, showing the range of stability of the globule over nanowire width W and applied field B, was calculated. It should be noted that the CUDA code of Filipp Rybakov was used instead of *Spirit* for wires with more than ca. 1 M spins.

A surprisingly large range of applied field B at which a single MG remains stable was found for nanowires of sufficient width W. When the applied field is increased and approaches $B = B_\mathrm{D}$ the edge modulations become suppressed, which limits the stability of the globule at high magnetic fields. Below a certain critical applied field, the Bloch points can escape from the system through the free edges, destroying the globule. However, it turns out that there is critical width of the nanowire of $W \approx 10L_\mathrm{D}$, above which the Bloch points may remain inside the sample even at zero magnetic field. After the magnetic field is increased again in the same direction, the globule forms anew.

Another significant point should be noted for this system: inside the region where the globule is metastable, there is a region where a single skyrmion tube along the wire is of lower energy than the conical phase, meaning that it represents the ground state of the system [212]. This is represented by the red region in Figure 4.9. Outside of this region, the skyrmion tube along the wire is metastable and at large enough field B it collapses, as indicated by the gray line in Figure 4.9. The fact that the skyrmion is energetically favourable in this region means that finite length skyrmion cores become more likely to be nucleated under thermal fluctuations. In the case that a long skyrmion tube is stabilised in the system, a way to create globules would be to cut the skyrmion tube, by thermal or other external excitation, such as a spatially localised alternating magnetic field or short pulse.

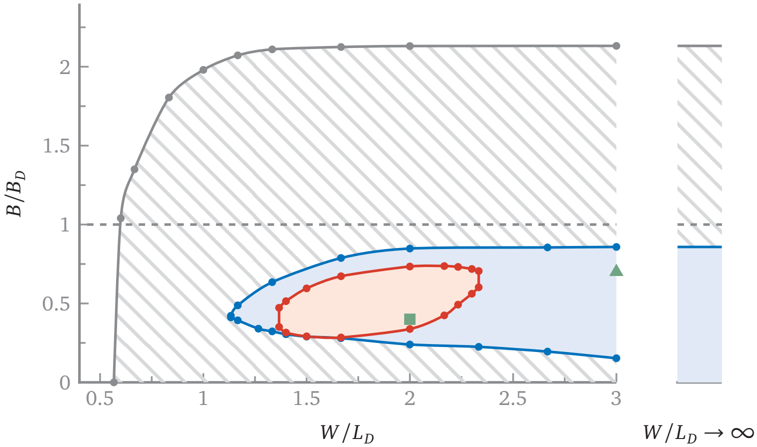


FIGURE 4.9 – Stability of the globule in an infinite nanowire

Stability diagram for the globule in the nanowire over wire width W and applied field B. The blue region indicates where the globule can be stabilised, where $W/L_\mathrm{D} \to \infty$ denotes convergence to a bulk system with free edges. The red region corresponds to the skyrmion tube along the wire being the ground state. The collapse field of the skyrmion tube is denoted by the gray line. The green triangle denotes parameters used to produce Figure 4.8 and the square denotes parameters used in Figure 4.10. This figure has been published in [61].

The question of how to create magnetic globules sets the direction of the next step. For this it is of interest to know the most favourable transition path for the nucleation of a globule out of a skyrmion tube, as well as the associated energy barriers. Figure 4.10 shows a MEP calculated for $W = 2L_\mathrm{D}$ and $H = 0.5H_\mathrm{D}$, where the single skyrmion tube is the ground state of the system ($E_\mathrm{a} < E_\mathrm{k}$). Note that the periodic boundary conditions along the nanowire result in a collapse path between state (e⟶i, dashed line) which does not have a strict physical meaning and may change with the domain length L_z.

The MEP clearly shows an energy barrier of $\Delta E = 4.7J$ for the collapse transition of the globule (i⟶k), which is on the same order of magnitude as the energy barrier for the chiral bobber estimated with the same method in Ref. [136] (see also Figure 4.3). The transitions e⟶c and c⟶a, corresponding to the annihilation of Bloch points, have significantly higher energy barriers of $\sim 30J$, which is consistent with the barriers in

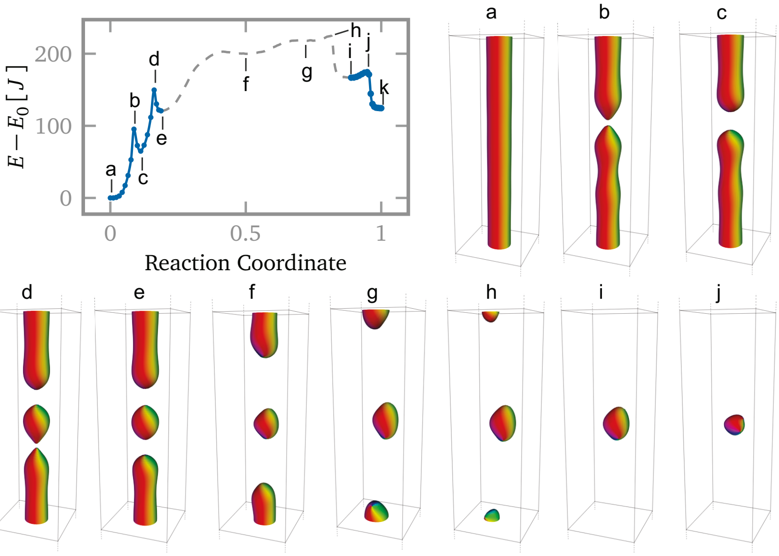


FIGURE 4.10 – Globule collapse GNEB Path

A nanowire of width $W = 2L_\mathrm{D}$ with periodical boundary conditions in z-direction at external field $B_\mathrm{ext} = 0.45B_\mathrm{D}$. The plot shows the MEP for the nucleation of a globule from a skyrmion tube by cutting the tube and nucleating two pairs of Bloch points, as well as the subsequent collapse of the globules. (a-j) show the $n_\mathrm{z} = 0$ isosurface at extremal points of the transition. (a) infinite skyrmion tube (b, d) saddle points for the nucleation of BP pairs, (c, e, i) stable configurations of coupled pairs of BPs. The intermediate part of the MEP, indicated as a dashed line, is an artefact of the periodical boundary and the retraction of the finite skyrmion tube into a globule, where local minima (f) and (g) appear along the path. The intermediate path is shown to illustrate that interactions between the BPs are nontrivial. This figure has been published in [61].

a thin layer shown in Figure 4.3. The consequence is that after a pair of BPs has been nucleated they will most likely remain in the sample rather than annihilate each other.

As the globule can evidently exist as a 3D-localised metastable state – especially due to its shape – one also needs to ask whether it would be possible to form a lattice of these objects and if they would cluster together on their own. For this, globule lattices were studied at as confined, as well as periodical structures, as shown in Figure 4.11.

As should be clear from the preceding results, a finite size geometry will allow globules to stabilize at the boundaries. Figure 4.11 shows that the presence of other globules can serve the same stabilising properties as the surface spirals and globules can exist in the center of such a system if the density of globules is high enough. Such states might be achieved by a fast annealing process from the paramagnetic phase.

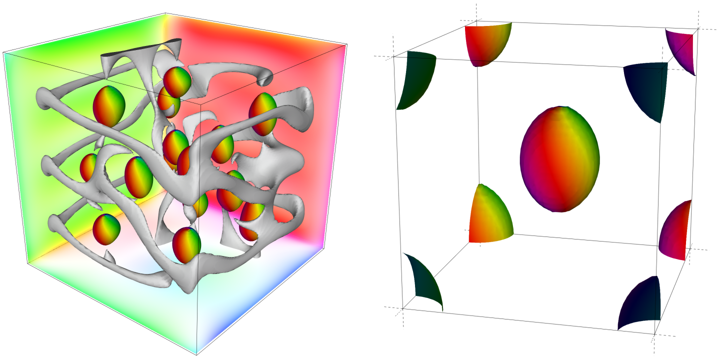


FIGURE 4.11 – Globule lattices

Lattices of globules in a confined space (left) and under periodical boundary conditions (right). On the left, the isosurfaces for $n_z = 0$ and $n_z = 1$ are shown, as well as three surfaces of the cube. Only the $n_z = 0$ isosurface is shown in the right image. The left image has been published in [61].

One might expect the globules to provide sufficient stabilisation to other globules, such that a cluster of them might become a metastable structure on its own. In order to test this, globules were placed in different lattice arrangements into a periodical system. Figure 4.12 shows the energy curves over cube size L for a set of external magnetic fields B for the simple cubic lattice of globules.

Stars indicate the collapse of the globules, meaning that the simple cubic lattice is absolutely unstable, as there is no local minimum on any of the curves, which would indicate an equilibrium lattice constant. This analysis was performed also for bcc, fcc and hcp arrangements of globules, but none showed an equilibrium lattice arrangement. While minima can appear in the energy curves, such as those presented in Figure 4.12, a variation of another lattice parameter always led to a further decrease of the energy, meaning there are plateaus, but no equilibrium lattice configurations.

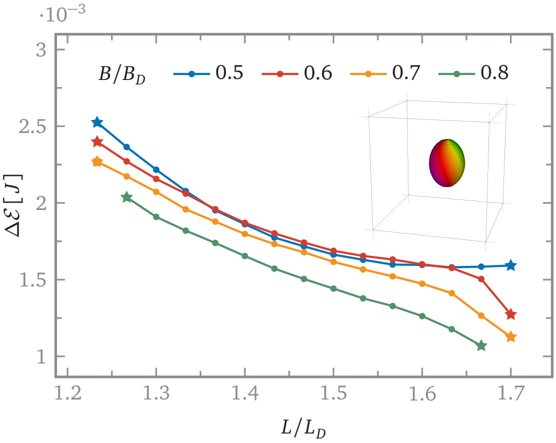


FIGURE 4.12 – Energy of the globule in a periodical cube

Simple cubic lattice of side length L. The plots shows the dependency of the difference in energy density $\Delta\mathcal{E}$ between a globule (see inset) and the conical phase ground state of the system) over the cube size L for a set of applied magnetic fields B. In this setup, L is also the period of the globule lattice. Stars indicate the points at which the globule collapses.

The falling energy with the lattice spacing L indicates that the globules want to be apart and will, space permitting, move away from each other, increasing the distance between them until they become unstable and collapse.

The calculations and images presented in this section give a good impression of the kind of complex 3D nanomagnetism, which can be tackled using *Spirit* and its GUI. Arguably, such systems cannot be studied to such an extent without real time visualisations of the simulated system or without running the numerically intensive parts on a GPU. Especially for very large systems of 100 M or more spins, optimisations for memory consumption and computational speed – such as in the CUDA code of Filipp Rybakov – become highly important.

The results themselves show that a wide class of isotropic chiral magnets can potentially host a novel, truly three-dimensional particle-like state, composed of a pair of Bloch points (BPs). The discovery of this object, named magnetic globule, extends the class of 3D-localised magnetic objects with particle-like properties and magnetic singularities coupled to a smooth magnetization field. By calculating MEPs using the GNEB method, the metastability of globules was proven by revealing finite energy barriers protecting it from collapsing and from growing.

It was shown that in a wide range of magnetic fields and geometries with both natural and artificial confinement, magnetic globules can be stabilized as isolated objects. While in homogeneous bulk systems the globule appears to be unstable, both on its own and as a cluster or lattice, both a skyrmion tube and a defect line in the conical background state can be nucleation points for single magnetic globules.

In the paper on this topic [61], suggestions for the experimental observability using high-resolution quantitative magnetic imaging methods are made. Off-axis electron holography [211, 213] in a transmission electron microscope in combination with tomography measurements, X-ray magnetic circular dichroism [214] or X-ray vector nanotomography [197] seem promising methods, but the techniques still have to evolve if coupled Bloch points are to be observed. An indirect detection might be achievable with magnetoresistive measurements, known to be sensitive to magnetic phase transitions in chiral magnetic nanostructures [209, 215].

It is to be expected that transport properties and dynamical behaviour of magnetic globules are highly non-trivial and may result in a number of new effects.

As the magnetic globule is consistently stabilised by the presence of inhomogeneities in the magnetic configuration, further research on these states could be performed in the context of defects and vacancies, which are known to modulate their surroundings. Similar to the nucleation of skyrmions, such points could be nucleation grounds for magnetic globules. Another conjecture in this context is the fact that the presence of Bloch points significantly increases the entropy of the configuration with respect to a regular skyrmion tube and could therefore be – together with the chiral bobber appearing at surfaces – an explanation for the A-phase in chiral magnets. Intuitively, the probability of nucleating a complete skyrmion string out of thermal fluctuation should shrink significantly when the sample size is increased. A finite length skyrmionic string – essentially a globule – seems far more likely to nucleate. It is possible that it is a lattice of coupled Bloch points, which constitutes the A-phase, or that globules are the structures from which skyrmions are nucleated when the Bloch points travel to the surfaces of the sample (analogous to what is described in [205]).

CONCLUSION 5

This work has been placed in the context of nanomagnetism, simulation of magnetisation dynamics and calculation of ground states and fundamental properties of systems and magnetic configurations. The atomistic spin model, used throughout this work, was introduced and illustrated. Due to the age of the most common software used for this model – generally written in Fortran – the implementation of a new C++ code was decided upon, which could be given a graphical user interface (GUI) and could be parallelised both on CPU and GPU systems. In Fortran, it is significantly more challenging to implement a cross-platform GUI or a GPU parallelisation, both of which usually require interaction with the C programming language.

Well-known methodology for atomistic spin systems – Monte Carlo (MC), energy minimisation and Landau-Lifshitz-Gilbert (LLG) dynamics – has been recounted in this thesis and the implementations in the new software *Spirit* have been verified against well-known results.

After covering recent, state-of-the-art methodological developments in the calculation of transition rates for magnetic systems, namely the geodesic nudged elastic band (GNEB) method and harmonic transition state theory (HTST), they were applied to skyrmions and skyrmionic states. It was shown that topology is neither a sufficient measure for the state of a skyrmionic configuration, nor a reliable argument when assessing the stability or especially the lifetime of skyrmions. HTST was shown to be a way of extracting physically realistic skyrmion lifetimes through the atomistic model with parameters from *ab initio* DFT calculations.

The need for knowledge of possible transitions of metastable magnetic configurations was discussed and the requirement of the HTST to know the possible transitions a metastable state may undertake has been laid out. As a solution to this issue, a new method for rate theory was developed, implemented in *Spirit* and applied to skyrmions: the minimum mode following (MMF) method. While this method has been known and used for molecular dynamics and atomic rearrangements, the way in which to transfer it to spin systems was previously unclear. In this work the mathematical basis was laid out and the successful application presented by revealing a previously undiscussed skyrmion transition: a mitosis-like duplication. It was further shown that this transition mechanism can be found in a wide range of parameters at which the skyrmion is stable and is therefore likely to be an important transition to take into account for skyrmion racetrack memories, where data persistence is paramount.

The *Spirit* software framework, developed in most part by the author of this work, has been presented. Its most important features have been shown, tests provided and applications and potential applications discussed. The code structure and design were explained and important implementation details highlighted, providing useful guidance for anyone wishing to modify *Spirit* or write a new code for spin systems.

Finally, the software was applied to complex 3D nanomagnetism and successfully revealed a novel magnetic state: the magnetic globule – two coupled Bloch points joined by a finite length cylindrical skyrmion tube. This demonstrated the versatility and usefulness of *Spirit* and the benefits of its advanced features for the study of complicated systems and structures.

Outlook

Numerous projects could build on this work, both in terms of implementation and application, and exciting research still lies ahead.

In addition to software-focused projects, such as bringing more of the methods implemented in *Spirit* to the Web user interface, several additional Hamiltonian models could be implemented, the two most relevant being the micromagnetic and the Ginzburg-Landau models [216, 217]. While the former could be implemented with ease, operating mostly on the same constraints, the latter would require considering of fluctuations of the spin length and would potentially be difficult to integrate into the design of *Spirit*. However, the implementation of these models would be of great benefit, as it would open new avenues for the simulation of macroscopic systems in the same software framework. Even more advanced implementations could combine the different models in the same system, for example by switching to an atomistic description in regions of strong magnetic curl, but for such concepts it seems to be necessary to implement a proper finite element method (FEM).

Further potential avenues are among others the non-collinear Alexander-Anderson model [76], the inclusion of longitudinal spin fluctuations (LSF) [218], for example via an additional term in the Hamiltonian describing how the total energy depends on the moment size [219, 220] or the addition of molecular dynamics for lattice degrees of freedom [104] to enable simultaneous modelling of magnetic and phononic low energy bosonic excitations in an adiabatic and atomistic regime using a unified theory based on ab initio DFT. Such a complete framework would enable the study of the interactions between the dynamics of the lattice and the magnetisation, which is relevant for magnon-phonon coupling [221, 222] and dynamical magneto-electric effects [223–227]. A good overview and more details are given in Ref. [63].

Numerous minor improvements can still be made, but they cannot be dealt with here. To cite but two: the implementation of an adaptive time step [228] in direct energy minimisation and LLG simulations, the addition of spin-spin correlation functions calculated with FFTs or improvements for antiferromagnetic or ferrimagnetic systems (e.g. using multiple interacting sublattices).

It has been discussed extensively how the calculation of lifetimes can be an important numerical tool, especially in the context of multiscale calculations. For the estimation of lifetimes, Langer's theory could be implemented in order to complement HTST, for which one would simply need to add the damping contributions to the current implementation of the dynamical matrix. This would enable quantitative comparisons of the methods and further studies of their respective benefits and limitations. The dynamical matrix could already be used for the visualisation of dynamical eigenmodes and magnon spectra, but the addition of the damping contribution would make such studies more complete.

Finally, the MMF and HTST methods both suffer from the size of the matrices needed for their calculations. Significant performance improvements could be made for example by using sparse matrices when dipolar interactions can be neglected. Parallelisation of the matrix diagonalisation used in both methods could also significantly speed up calculations, but large systems would still be memory-bound.

Further calculations based on the numerical results of this work would also be of interest and it is obvious that the development of a way to apply HTST to larger systems would be highly useful. The higher order skyrmionic states discussed in this work could be analysed with respect to their lifetime, though it is an open question how to accurately calculate the volume of the saddle points for the breaking of a skyrmionic "sack". The author expects a limitation of the lifetimes of higher order skyrmionic states due to the increasing number of saddle points through which a "sack" or loop might break or a skyrmionic core might collapse or collide with another core or the wall, while the energy barriers do not appear to significantly depend on the size of the structure and are therefore not expected to have much influence.

Evidently, for these states, the number of possible transitions quickly increases quite dramatically and the skyrmion duplication/collision transition, originally discovered by the MMF method, is a key component. The calculation of transition rates for this specific transition would be of more general interest, as it might be engineered as a nucleation mechanism for skyrmions or become a limiting factor in data persistence in a skyrmion racetrack, where information can be changed by the thermally induced nucleation of a skyrmion. Inversely, the collision of two skyrmions might be a limiting factor when driving the data along the track. When one skyrmion is slightly pinned at a defect, a following one might collide and merge with it, again leading to data corruption.

Since the discovery of the chiral magnetic bobber, it has been suggested for an alternative skyrmion racetrack, where the skyrmion and bobber might represent the "0" and "1" bits respectively. Therefore, the simulation of such a device is of interest and the current-driven dynamics of the bobber should be researched in order to assess the feasibility of such a device design. The study of the properties of Bloch points (BPs) plays an important role in the understanding of complex 3D systems and the discovery of the magnetic globule as a metastable configuration opens up more questions about their ability to couple and their behaviour under spin torques. The fundamental study of energy eigenmodes and dynamical eigenmodes in such highly non-collinear systems, in dependence on parameters and present states inside the magnetic configuration, could reveal important properties and lead to new discoveries.

In summary, the developments in methodology and software and the discoveries about chiral magnetic states achieved by this work lay a broad and solid foundation for future research in magnetism. This work enables the study of complex magnetic textures and nanoscale states to be pushed a significant step further.

APPENDIX

A ENERGY DATA INTERPOLATION

Energy paths throughout this work, such as shown in Figure 1.8, Figure 2.9 or Figure 4.2, have been interpolated between the discrete images using cubic Hermite polynomials. This approach has been previously described, for example in Refs. [56] and [108]. The images themselves are generally represented as dots on the line, while the line describes the interpolated values. This is visualised schematically in Figure A.1, where it is also shown how the interpolation may reveal intermediate minima or the non-convergence of a path.

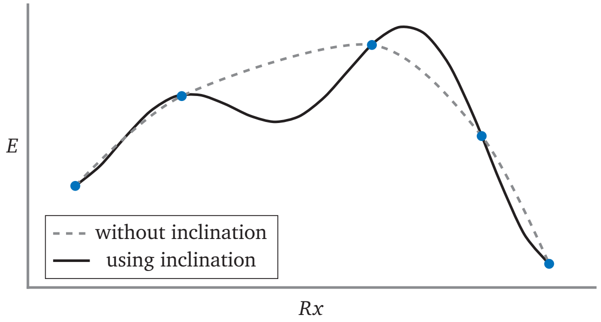


FIGURE A.1 – Interpolation with hermite polynomials

Schematic of interpolation of energy data points (blue). The piecewise interpolation with Hermite polynomials **??**, using the inclination information, (solid line) may show more features and provide more information than a spline fit without information about the inclinations (dashed line). It is shown schematically, how an intermediate minimum may go unnoticed if not using the inclinations. Image adapted from [108].

Viewing the interpolated energies between the data points during or after a GNEB calculation – as can be done in the GUI of *Spirit* (see Figure 3.12) – allows checking the convergence of a climbing or falling image or to better judge the general quality of convergence of a transition path.

As, during a GNEB calculation, the distance $L^{\nu\mu}$ in configuration space, as well as the energy difference $\Delta E^{\mu\nu}$, between two images M^{μ} and M^{ν} are known, as well as the energy gradients at those images, the interpolations can be efficiently calculated as byproducts of the calculation.

In the following, the data points will be denoted p^{ν} and the inclinations with respect to an order parameter ξ^{ν} will be denoted q^{ν}. Taking the reaction coordinate as the order parameter, $\xi^{\nu} = Rx$, one need only determine the two quantities

$$p^{\nu} = E^{\nu}(Rx) \qquad \text{A.1}$$

and

$$q^\nu = ||\nabla E^\nu(Rx)|| = -H^\nu_{\text{eff}}(Rx) \cdot \hat{\tau}^\nu_{\mathcal{T}}(Rx)\,. \tag{A.2}$$

In the interval $(Rx^\nu, Rx^{\nu+1})$, the piecewise interpolating cubic Hermite polynomials are parameterised as

$$c(t) = h_{00}(t)p^\nu + h_{10}(t)p^{\nu+1} + h_{01}(t)q^\nu d + h_{11}(t)q^{\nu+1}d\,, \tag{A.3}$$

where

$$t \in [0,1] \quad \text{and} \qquad d = Rx^{\nu+1} - Rx^\nu = L^{\nu,\nu+1} \tag{A.4}$$

and the functions h are defined as

$$\begin{aligned} h_{00} &= 2t^3 - 3t^2 + 1\,,\\ h_{10} &= -2t^3 + 3t^2\,,\\ h_{01} &= t^3 - 2t^2 + t\,,\\ h_{11} &= t^3 - t^2\,. \end{aligned} \tag{A.5}$$

This interpolation scheme may also be applied to an arbitrary other order parameter, such as the z-component of the magnetisation ($\xi = M_z$). To apply the piecewise interpolation, a set of points ξ^ν and inclinations $d\xi^\nu/dRx$ is required. Using the chain rule, the inclination can be expressed as

$$\frac{d\xi^\nu}{dRx} = \frac{d\xi^\nu}{dE}\frac{dE^\nu}{dRx}\,. \tag{A.6}$$

Since the energy gradient along the path can easily be calculated by projecting it on the normalised path tangent τ_ν as

$$q^\nu = \frac{dE^\nu}{dRx} = -H^\nu_{\text{eff}} \cdot \tau^\nu\,. \tag{A.7}$$

If $d\xi^\nu/dE$ is known or can be calculated by finite differences, the interpolation can then be calculated. In the case of $\xi^\nu = M^\nu_z$, one may use

$$\frac{dM^\nu_z}{dRx} = \sum_i \frac{dM_{z,i}}{dE}\frac{dE}{dRx} = \sum_i \frac{1}{dE/dM_{z,i}}\frac{dE}{dRx} = -\sum_i \frac{1}{H^{\text{eff}}_{z,i}}\frac{dE}{dRx}\,. \tag{A.8}$$

Thus, the energy values can in principle, with the help of the reaction coordinate, be interpolated over various useful order parameters.

B GAUSSIAN HAMILTONIAN

B.1 *Derivatives*

This appendix will deal with the interaction-free Gaussian Hamiltonian and its derivatives. We recall the Hamiltonian (1.17) is defined – per spin – as

$$H = \sum_i H_i = \sum_i a_i \exp\left(-\frac{l_i^2(\vec{n})}{2\sigma_i^2}\right), \tag{B.1}$$

where a_i is the amplitude, σ_i the width and $l_i(\vec{n})$ is a measure of distance of the spin $\vec{n}$ to the center $\vec{c}_i$ of the ith gaussian function. This yields the gradient

$$\nabla H = -\sum_i a_i \exp\left(-\frac{l_i^2(\vec{n})}{2\sigma_i^2}\right), \frac{l_i(\vec{n})}{\sigma_i^2}\nabla l_i\,, \tag{B.2}$$

which can be written in index notation as

$$\partial_\alpha H = -\sum_i a_i \exp\left(-\frac{l_i^2(\vec{n})}{2\sigma_i^2}\right)\frac{l_i(\vec{n})}{\sigma_i^2}\partial_\alpha l_i(\vec{n})\,, \tag{B.3}$$

which in turn gives us the Hessian

$$\begin{aligned} \mathrm{Hess}(H) &= \partial_\alpha\partial_\beta H \\ &= \sum_i a_i \exp\left(-\frac{l_i^2(\vec{n})}{2\sigma_i^2}\right) \\ &\quad \frac{1}{\sigma_i^2}\left[\left(\frac{l_i(\vec{n})^2}{\sigma_i^2}-1\right)\partial_\beta l_i(\vec{n})\partial_\alpha l_i(\vec{n}) - l_i(\vec{n})\partial_\beta\partial_\alpha l_i(\vec{n})\right]. \end{aligned} \tag{B.4}$$

The geodesic distance between the vectors was purposely not chosen as the geodesic measure of distance, but as

$$l_i(\vec{n}) = 1 - \vec{n}\cdot\vec{c}_i\,, \tag{B.5}$$

for which the derivatives are

$$\partial_\alpha l_i = -c_{i\alpha}, \quad \partial_\beta\partial_\alpha l_i = 0\,. \tag{B.6}$$

The gradient becomes

$$\nabla H = \sum_i a_i \exp\left(-\frac{(1-\vec{n}\vec{c}_i)^2}{2\sigma_i^2}\right)\frac{(1-\vec{n}\vec{c}_i)}{\sigma_i^2}\vec{c}_i\,. \tag{B.7}$$

The Hessian becomes

$$\mathrm{Hess}(H) = \sum_i a_i \exp\left(-\frac{(1-\vec{n}\vec{c}_i)^2}{2\sigma_i^2}\right)\frac{1}{\sigma_i^2}\left(\frac{(1-\vec{n}\vec{c}_i)^2}{\sigma_i^2}-1\right)c_{i\alpha}c_{i\beta}\,. \tag{B.8}$$

B.2 GNEB test script

The following Python script shows how, at time of writing this work, the Python API of *Spirit* can be used to reproduce the data of the GNEB calculation shown in Figure 1.7. Note that this is merely an example of how the API could be used.

```python
from spirit import state
from spirit import system
from spirit import chain
from spirit import simulation
from spirit import configuration
from spirit import hamiltonian
from spirit import io
from spirit import geometry
from spirit import parameters
from spirit import transition
from spirit import quantities
from spirit import constants

import numpy as np
import os

#######
output_folder = os.path.join(scriptdir, "output")
if not os.path.exists(output_folder):
os.makedirs(output_folder)
input_file = os.path.join(scriptdir, "input.cfg")
#######

LLG = simulation.METHOD_LLG
GNEB = simulation.METHOD_GNEB
VP = simulation.SOLVER_VP
OVF_TXT = io.FILEFORMAT_OVF_TEXT

LANDSCAPE_RESOLUTION = 100

NOI = 6

with state.State(input_file, quiet=True) as p_state: # State setup
    # Set parameters
    parameters.llg.set_output_general(p_state, any=False) # Disallow any output
    parameters.llg.set_convergence(p_state, 1e-12)
    parameters.gneb.set_output_general(p_state, any=False) # Disallow any output
    parameters.gneb.set_convergence(p_state, 1e-12)
    parameters.gneb.set_n_interpolation(p_state, 20)

    # Generate data of energy landscape over theta and phi
    print("--- sampling the energy landscape")
    with open(os.path.join(output_folder, "landscape.txt"), "w") as outfile:
        for theta in np.linspace(0.001, np.pi-0.001, LANDSCAPE_RESOLUTION):
            for phi in np.linspace(0.001-np.pi, np.pi-0.001, LANDSCAPE_RESOLUTION):
                spherical = [theta, phi]
                direction = [np.sin(theta)np.cos(phi), np.sin(theta)np.sin(phi), np.cos(theta)]
                configuration.domain(p_state, direction)
                system.update_data(p_state)
                energy = system.get_energy(p_state)
                outfile.write("{:^20e} {:^20e}\n".format(' '.join(map(repr, spherical)),
                    str(energy)))
    print("--- energy landscape sampling done")

    # Generate chain of images, write output and perform GNEB calculations
    print("--- gneb calculation...")

    print("--- creating first image")
    configuration.domain(p_state, [0, -1, 0])
    configuration.add_noise(p_state, 1)
    io.image_write(p_state, os.path.join(output_folder, "image_first_initial.ovf"),
        fileformat=OVF_TXT)
    simulation.start(p_state, LLG, VP, n_iterations=100000, n_iterations_log=100000)
    io.image_write(p_state, os.path.join(output_folder, "image_first.ovf"), fileformat=OVF_TXT)

    print("--- changing chain length")
    chain.image_to_clipboard(p_state)
    chain.set_length(p_state, NOI)

    print("--- creating final image")
    configuration.domain(p_state, [0, 1, 0], idx_image=NOI-1)
    configuration.add_noise(p_state, 1, idx_image=NOI-1)
    io.image_write(p_state, os.path.join(output_folder, "image_last_initial.ovf"),
        fileformat=OVF_TXT, idx_image=NOI-1)
    simulation.start(p_state, LLG, VP, n_iterations=100000, n_iterations_log=100000,
        idx_image=NOI-1)
    io.image_write(p_state, os.path.join(output_folder, "image_last.ovf"), fileformat=OVF_TXT,
        idx_image=NOI-1)

    # Initial Path
    print("--- interpolating")
    transition.homogeneous(p_state, 0, NOI-1)
    chain.update_data(p_state)
    io.chain_write(p_state, os.path.join(output_folder, "chain_initial.ovf"), fileformat=OVF_TXT)
    Rx = chain.get_reaction_coordinate(p_state)
    energies = chain.get_energy(p_state)
    with open(os.path.join(output_folder, "energies_initial.txt"), "w") as outfile:
        for i in range(len(Rx)):
            outfile.write("{:^20e} {:^20e}\n".format(Rx[i], energies[i]))
```

```
Rx_interp = chain.get_reaction_coordinate_interpolated(p_state)
energies_interp = chain.get_energy_interpolated(p_state)
with open(os.path.join(output_folder, "energies_initial_interp.txt"), "w") as outfile:
    for i in range(len(Rx_interp)):
        outfile.write("{:^20e} {:^20e}\n".format(Rx_interp[i], energies_interp[i]))

# First GNEB calculation
print("——— first relaxation")
simulation.start(p_state, GNEB, VP, n_iterations=100000, n_iterations_log=100000)
io.chain_write(p_state, os.path.join(output_folder, "chain_relaxed.ovf"),
    fileformat=OVF_TXT)
Rx = chain.get_reaction_coordinate(p_state)
energies = chain.get_energy(p_state)
with open(os.path.join(output_folder, "energies_relaxed.txt"), "w") as outfile:
    for i in range(len(Rx)):
        outfile.write("{:^20e} {:^20e}\n".format(Rx[i], energies[i]))
Rx_interp = chain.get_reaction_coordinate_interpolated(p_state)
energies_interp = chain.get_energy_interpolated(p_state)
with open(os.path.join(output_folder, "energies_relaxed_interp.txt"), "w") as outfile:
    for i in range(len(Rx_interp)):
        outfile.write("{:^20e} {:^20e}\n".format(Rx_interp[i], energies_interp[i]))

# GNEB with climbing and falling images (set automatically)
print("——— CI relaxation")
parameters.gneb.set_image_type_automatically(p_state)
simulation.start(p_state, GNEB, VP, n_iterations=100000, n_iterations_log=100000)
io.chain_write(p_state, os.path.join(output_folder, "chain_relaxed_ci.ovf"),
    fileformat=OVF_TXT)
Rx = chain.get_reaction_coordinate(p_state)
energies = chain.get_energy(p_state)
with open(os.path.join(output_folder, "energies_relaxed_ci.txt"), "w") as outfile:
    for i in range(len(Rx)):
        outfile.write("{:^20e} {:^20e}\n".format(Rx[i], energies[i]))
Rx_interp = chain.get_reaction_coordinate_interpolated(p_state)
energies_interp = chain.get_energy_interpolated(p_state)
with open(os.path.join(output_folder, "energies_relaxed_ci_interp.txt"), "w") as outfile:
    for i in range(len(Rx_interp)):
        outfile.write("{:^20e} {:^20e}\n".format(Rx_interp[i], energies_interp[i]))

print("——— gneb calculation done")
```

LISTING 1 –

Python script for single spin GNEB test case

B.3 *MMF test script*

The following Python script shows how, at time of writing this work, the Python API of *Spirit* can be used to reproduce the data of the paths of the MMF calculation shown in Figs. 2.4–2.6. Note that this is merely an example of how the API could be used.

```python
import os
import subprocess
from subprocess import CalledProcessError

import numpy as np

### Import spirit library
from spirit import state
from spirit import system
from spirit import hamiltonian
from spirit import geometry
from spirit import chain
from spirit import configuration
from spirit import transition
from spirit import simulation
from spirit import quantities
from spirit import io
from spirit import log

datafolder_fixed = os.path.abspath(os.path.join(os.path.dirname( __file__ ), "output_fixed3"))

cfgfile = "single_spin_fixed3.cfg"

with state.State(cfgfile) as p_state:
    for n in range(40):
        # Start from the local minimum close to −z
        configuration.MinusZ(p_state)
        simulation.PlayPause(p_state, "LLG", "VP")
        # Randomise the starting position
        configuration.Add_Noise_Temperature(p_state, 0.2)
        # Run the MMF calculation
        if (n>=0):
            simulation.PlayPause(p_state, "MMF", "VP")

        # Move the data from the fixed output file to a custom name
        fname_in = datafolder_fixed+"/mm_vf_Image−−1_Spins−archive.txt"
        fname_out = datafolder_fixed+"/mmf_path_{}.txt".format(n)
        try:
            subprocess.run("mv {} {}".format(fname_in, fname_out), shell=True, check=True)
        except CalledProcessError as error:
            print("Error moving file. Message: ", error.output)
```

LISTING 2 –

Python script for single spin MMF test case

C DERIVATIONS OF THE HESSIAN MATRIX ON RIEMANNIAN MANIFOLDS

C.1 *The spin manifold*

As was mentioned for example in Sections I.2, I.4, 1.2 and 2.2, the spins are constrained to the physical manifold, given by Equation (I.8):

$$\mathcal{M} = \bigotimes_{i=1}^{N} \mathbb{S}^2 \subset \mathbb{R}^{3N} \,.$$

It is a Riemannian submanifold of the embedding euclidean space $\mathcal{E} = \mathbb{R}^{3N}$. As before, we denote smooth extensions to the embedding space $\mathcal{E}$ with a bar, for example a function f on $\mathcal{M}$ would have the smooth extension $\bar{f}$.

The atomistic Heisenberg Hamiltonian (I.1) can be written very generally as a combination of linear and quadratic contributions in $\vec{n}$:

$$\mathcal{H} = L^i_\alpha n^i_\alpha + n^i_\alpha Q^{ij}_{\alpha\beta} n^j_\beta \,. \tag{C.1}$$

This easily gives the derivatives in the embedding space:

$$\frac{\partial \bar{\mathcal{H}}}{\partial n^i_\alpha} = L^i_\alpha + Q^{ij}_{\alpha\beta} n^j_\beta + n^j_\beta Q^{ji}_{\beta\alpha} \,, \tag{C.2}$$

$$\frac{\partial^2 \bar{\mathcal{H}}}{\partial n^i_\alpha \partial n^j_\beta} = Q^{ij}_{\alpha\beta} + Q^{ji}_{\beta\alpha} \,. \tag{C.3}$$

As noted in the main part of this work, in general the second derivatives of the energy with respect to the spin orientations $\partial^2 \mathcal{H}$ need to be performed covariantly. The following sections will detail some approaches alternative to the projector approach, using the so-called Weingarten map, chosen in this thesis and presented in Chapter 2.

C.2 *Projection*

We can project any vector or matrix from the embedding space $\mathcal{E}$ (in euclidean representation) onto the tangent space to the manifold $\mathcal{M}$ at a point $x \in \mathcal{M}$ using a projector $P(x)$. For the sphere it is known the matrix representation in index notation reads

$$P^{ij}_{\alpha\beta} = \delta_{ij}(\delta_{\alpha\beta} - x^i_\alpha x^j_\beta) \,. \tag{C.4}$$

Compare this also to (2.5), where it is used that $P^\perp_x = I - P_x = x^T x$.

It is worth noting that the first covariant derivatives correspond simply to a projection onto the tangent space

$$\partial\mathcal{H} = P\partial\bar{\mathcal{H}}\,, \tag{C.5}$$

allowing to use euclidean coordinates to effortlessly calculate the covariant energy gradients.

C.3 *Transformation*

In order to (basis-) transform vectors between the $(3N)$ euclidean basis and the $(2N)$ tangent space basis we simply need a matrix composed of two orthonormal tangent vectors, represented in the embedding space (i.e. $3N$) coordinates. *These tangent vectors may be generated from for example a) random vectors and subsequent orthogonalization b) a right-handed triplet and subsequent orthogonalization, or - and this is probably the preferable choice - c) the unit vectors of spherical coordinates θ and φ.* Choosing the latter, we can write the transformation matrix

$$\begin{aligned} T &= \{\vec{e}_\theta, \vec{e}_\varphi\} \\ &= \begin{pmatrix} \cos\theta\cos\varphi & -\sin\varphi \\ \cos\theta\sin\varphi & \cos\varphi \\ -\sin\theta & 0 \end{pmatrix} \\ &= \begin{pmatrix} zx/r_{xy} & -y/r_{xy} \\ zy/r_{xy} & x/r_{xy} \\ -r_{xy} & 0 \end{pmatrix}, \end{aligned} \tag{C.6}$$

where $r_{xy} = \sin\theta = \sqrt{1-z^2}$. This matrix is even reasonably inexpensive to compute. Note that the poles need to be excluded, but one may simply choose $T = \{\vec{e}_x, \vec{e}_y\}$ and $T = \{\vec{e}_x, -\vec{e}_y\}$ at the two poles respectively, and re-orthodonalise the vectors with respect to the spin orientation.

For the transormation of matrices, $A^{\mathrm{2N}} = T^T A^{\mathrm{3N}} T$ works fine but the inverse transformation is not known to the author, as $T^T T = \mathbb{1}|^{2N}$ but $T T^T \neq \mathbb{1}$. Note also that the transformation T of course automatically removes non-tangential components so that

$$T^T P = T^T\,. \tag{C.7}$$

C.4 *Constrained Hessian approach*

In some cases, such as HTST calculations (see Section 1.3), we only need to calculate the Hessian at stationary points, at which the covariant energy gradient vanishes (i.e. the spins are parallel to the effective field). The following describes a straightforward derivation for such cases.

We write the Lagrangian

$$L = \bar{\mathcal{H}} - \sum_i \lambda^i (x^i_\alpha x^i_\alpha - 1) . \tag{C.8}$$

By assuming that we are at a local extremum, we can calculate the Lagrange multipliers

$$\begin{aligned} & \frac{\partial L}{\partial x^i_\alpha} = 0 \\ \Rightarrow \quad & \frac{\partial \bar{\mathcal{H}}}{\partial x^i_\alpha} - \lambda^i 2 x^i_\alpha = 0 \\ \Rightarrow \quad & \lambda^i = \frac{1}{2} x^i_\alpha \nabla^i_\alpha \bar{\mathcal{H}} \end{aligned} \tag{C.9}$$

The constrained Hessian is then written

$$\begin{aligned} H &= \frac{\partial^2 L}{\partial x^i_\alpha \partial x^i_\beta} \\ &= \frac{\partial^2 \bar{\mathcal{H}}}{\partial x^i_\alpha \partial x^i_\beta} - 2\delta_{ij}\delta_{\alpha\beta}\lambda^i \\ &= \bar{H} - \delta_{ij}\delta_{\alpha\beta} x^i_\alpha \nabla^i_\alpha \bar{\mathcal{H}} , \end{aligned} \tag{C.10}$$

which in our notation becomes the Hessian in $3N$-representation

$$H = \bar{H} - \mathbb{1}(x^j \cdot \nabla^j \bar{\mathcal{H}}) . \tag{C.11}$$

Due to the fact that $T^T T = \mathbb{1}$ the $2N$-representation reads

$$H|^{2N} = T^T H T = T^T \bar{H} T - \mathbb{1}(x^j \cdot \nabla^j \bar{\mathcal{H}}) . \tag{C.12}$$

C.5 *Projector-based approach*

This approach is more general and is very close to what has been used for the MMF method in this thesis (see Chapter 2). The result is the same, but the derivation more intuitive and, however, less rigorous.

We denote the manifold $\mathcal{M}$ and define the tangential gradient

$$\nabla = P \bar{\nabla} , \tag{C.13}$$

where P is the projection operator, taking a vector to the tangent space of $\mathcal{M}$ in any point $x \in \mathcal{M}$ and ∇ and $\bar{\nabla}$ denote the derivative of any function on $\mathcal{M}$ and the derivative of the same function on the embedding space $\mathcal{E}$. Compare this also to Equation C.5.

From $H = \nabla\nabla$ we can now easily define the tangential Hessian

$$H = \nabla\nabla = P\bar{\nabla}P\bar{\nabla}\,. \tag{C.14}$$

With the chain rule, this gives

$$H^{ij}_{\alpha\beta} = (P\bar{H}P^T)^{ij}_{\alpha\beta} + P^{im}_{\alpha\mu}(\bar{\nabla}^m_\mu P^{jn}_{\beta\nu})\bar{\nabla}^n_\nu\,. \tag{C.15}$$

On spin systems, i.e. the 2N-sphere:

We will use ($\cdot$) and ($\otimes$) as the inner and outer product of vectors respectively, as well as ($\oplus$) as the direct sum of vector spaces. We now consider a spin system, where the spins are constrained to constant length. The manifold on which the system is allowed to move is the $2N$-dimensional direct product of N spheres:

$$M = \bigotimes_{i=1}^{N} S^2 \subset \mathbb{R}^{3N}\,. \tag{C.16}$$

Trivially, any 3-vector can be projected into the tangent plane to a surface by removing the normal component. In this case, the normals are equal to the respective point on the sphere and thus the projector takes a block-diagonal form:

$$P = \bigoplus_{i}^{N} P^i_{\alpha\beta} = \bigoplus_{i}^{N}(\delta_{\alpha\beta} - x^i_\alpha x^i_\beta)\,, \tag{C.17}$$

where $\oplus$ denotes the direct sum of the matrices over the spin index $\nu \in [1..N]$. Note that this projector fulfills $P = P^T$. In index notation we can write this using δ_{ij}:

$$P^{ij}_{\alpha\beta} = \delta_{ij}(\delta_{\alpha\beta} - x^i_\alpha x^j_\beta)\,. \tag{C.18}$$

After some derivations we end up with an expression for the Hessian in Euclidean (i.e. $3N$-) representation:

$$H^{ij} = P^{ii}\bar{H}^{ij}P^{jj} - P^{ij}(x^j \cdot \bar{\nabla}^j) - (P^{ij}\bar{\nabla}^j) \otimes x^j\,. \tag{C.19}$$

Note that the resulting $3N \times 3N$ Hessian matrix can be transformed into the $2N \times 2N$ basis of the tangent space to significantly reduce time needed to calculate eigenmodes. This gives the following:

$$H^{ij} = T^T\bar{H}^{ij}T - T^T(x^j \cdot \bar{\nabla}^j)T - (T^T\bar{\nabla}^j) \otimes x^j T\,, \tag{C.20}$$

which, as $T^T T = \mathbb{1}$ and $x^j T = 0$, results in

$$H|^{2N} = T^T H T = T^T\bar{H}T - \mathbb{1}(x^j \cdot \bar{\nabla}^j)\,. \tag{C.21}$$

Interestingly, this corresponds exactly to the result one gets when writing a constrained Hessian using Lagrange multipliers.

C.6 *Spherical coordinate approach*

Another method to calculate the second derivatives at stationary points, i.e. where the covariant gradient $\partial\mathcal{H}$ vanishes, is to use spherical coordinates. We denote spherical coordinates $\xi_\alpha \in \{\theta, \phi\}$. We write the gradient in spherical coordinates as

$$\begin{aligned}\nabla_{\xi_\alpha}\mathcal{H}((x,y,z)(\theta,\phi)) &= \frac{\partial\mathcal{H}(\theta,\phi)}{\partial\xi_\alpha}\\ &= \frac{\partial\bar{\mathcal{H}}(x,y,z)}{\partial x_\beta}\frac{\partial x_\beta}{\partial\xi_\alpha}\\ &= \nabla_{x_\beta}\bar{\mathcal{H}}(x,y,z)\tilde{J}_{\beta\alpha}\,,\end{aligned} \tag{C.22}$$

where $\tilde{J}$ is the Jacobian of the coordinate transformation

$$\tilde{J}^T = \tilde{J}_{\beta\alpha} = \frac{\partial x_\beta}{\partial\xi_\alpha}\,. \tag{C.23}$$

The Hessian in spherical coordinates is then

$$\begin{aligned}H(\mathcal{H}(\theta,\phi)) &= \frac{\partial^2\mathcal{H}(\theta,\phi)}{\partial\xi_\alpha\partial\xi_\beta}\\ &= \frac{\partial\nabla_{x_\beta}\bar{\mathcal{H}}\tilde{J}_{\beta\alpha}}{\partial\xi_\alpha}\\ &= \nabla_{x_\alpha}\nabla_{x_\beta}\bar{\mathcal{H}}\tilde{J}_{\beta\alpha}J_{\beta\alpha} + \nabla_{x_\beta}\bar{\mathcal{H}}\tilde{H}_{\alpha\beta}\,,\end{aligned} \tag{C.24}$$

where $\tilde{H}$ here denotes the Hessian of the coordinate transformation

$$\tilde{H} = \frac{\partial\tilde{J}}{\partial\xi_\alpha} = \frac{\partial x_\gamma}{\partial\xi_\alpha\partial\xi_\beta}\,. \tag{C.25}$$

C.7 *Covariant approach*

The covariant treatment of second derivatives can also be done in spherical coordinates, instead of a projector approach in the embedding space. However, we will see that this approach suffers from divergencies at the poles of the spherical coordinates – the usual problem – and is therefore impractical for our means.

In this formulation we get correction terms accounting for the curvature of the underlying manifold:

$$H_{\text{cov}}(\mathcal{H}) = \frac{\partial^2\mathcal{H}(\theta,\phi)}{\partial\xi_\alpha\partial\xi_\beta} - \Gamma^\alpha_{jk}\nabla_{\xi_\alpha}\mathcal{H}\,, \tag{C.26}$$

which we can easily rewrite in euclidean representation. The Christoffel symbols Γ for spherical coordinates are well-known and easily calculated.

The Hessian can be written almost without use of trigonometric functions, with derivatives calculated entirely in $3N$:

$$\begin{aligned}
H_{\text{cov}}(\mathcal{H}) =& \frac{\partial^2 \mathcal{H}(\theta,\phi)}{\partial \xi_\alpha \partial \xi_\beta} - \Gamma^i_{jk} \nabla_{\xi_\alpha} \mathcal{H} \\
=& \nabla_{x_\alpha} \nabla_{x_\beta} \bar{\mathcal{H}} \tilde{J}_{\beta\alpha} \tilde{J}_{\beta\alpha} \\
& + \nabla_{x_\beta} \bar{\mathcal{H}} \tilde{H}_{\alpha\beta} - \Gamma^i_{jk} P \nabla_{x_\alpha} \bar{\mathcal{H}} \\
=& \tilde{J} H_x(\bar{\mathcal{H}}) \tilde{J} \\
& + \text{grad}_x(\bar{\mathcal{H}}) \tilde{H} - \Gamma^\alpha_{jk} (\tilde{J} \text{grad}_x(\bar{\mathcal{H}}))_\alpha \,,
\end{aligned} \tag{C.27}$$

where $\tilde{J}$ and $\tilde{H}$ are again the Jacobian and Hessian of the spherical coordinate transformation and $\tilde{J}$, $\tilde{H}$ and Γ can all be efficiently written in euclidean coordinates:

$$\Gamma^\theta = \begin{pmatrix} 0 & 0 \\ 0 & -\sin\theta\cos\theta \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 0 & -z r_{xy} \end{pmatrix}, \tag{C.28}$$

$$\Gamma^\varphi = \begin{pmatrix} 0 & \cot\theta \\ \cot\theta & 0 \end{pmatrix} = \begin{pmatrix} 0 & z/r_{xy} \\ z/r_{xy} & 0 \end{pmatrix}, \tag{C.29}$$

where we merely need to exclude the poles of the sphere. Note that this may produce unexpected behaviour, if one does not transform into a different spherical coordinate system beforehand, which is why this approach is not the most practical.

D MMF MODE AND FORCE FIELDS

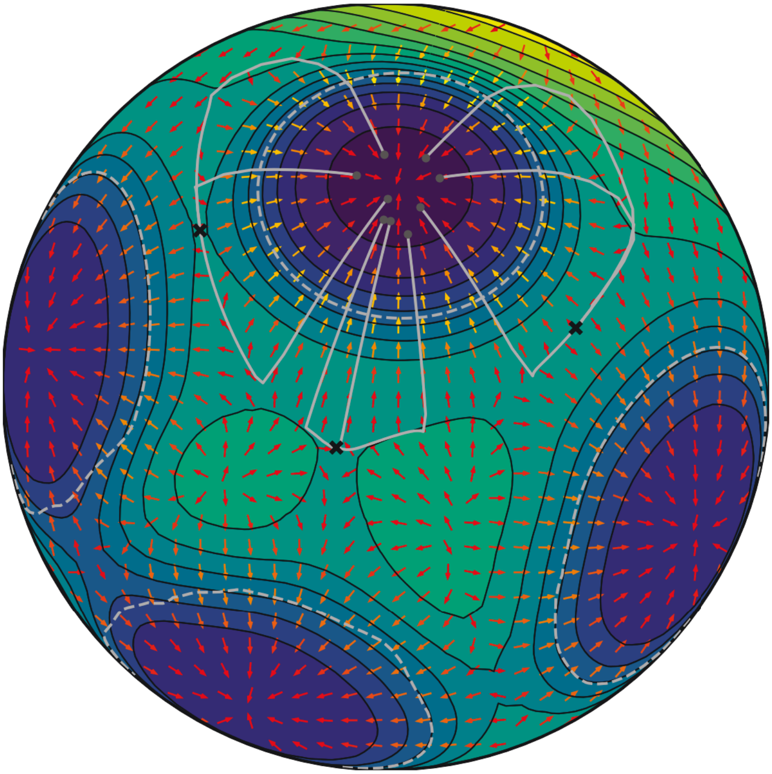


FIGURE D.1 – Spherical plot of the gradient force in the MMF method

A single spin in a superposition of Gaussian potentials (see also Section 1.2, Chapter 2 and Appendix Section B). As the configuration space manifold $\mathcal{M}$ is a sphere (see Section I.2), the energy landscapecan be plotted, as well as the MMF search paths and force fields. The gradient force is shown, pointing from higher energies (yellow, green) to the local minima (blue).

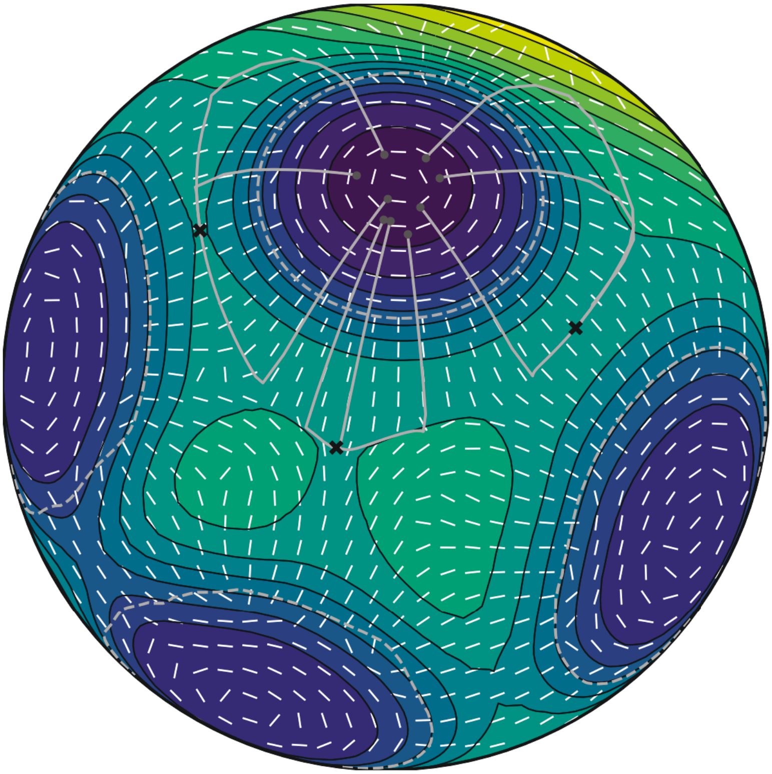


FIGURE D.2 – Spherical plot of the mode followed in the MMF method

A single spin in a superposition of Gaussian potentials (see also Section 1.2, Chapter 2 and Appendix Section B). As the configuration space manifold $\mathcal{M}$ is a sphere (see Section I.2), the energy landscapecan be plotted, as well as the MMF search paths and force fields. The minimum mode is shown, indicating the symmetry of the energy landscape and illustrating how the minimum mode points orthogonal to the energy contour line at the saddle point. Note that the gray dashed lines encircle the convex regions of the energy landscape, inside which the minimum mode does not necessarily have any usefulness for the MMF method, as following for example a circular mode around a local minimum would not lead the system out of the convex region.

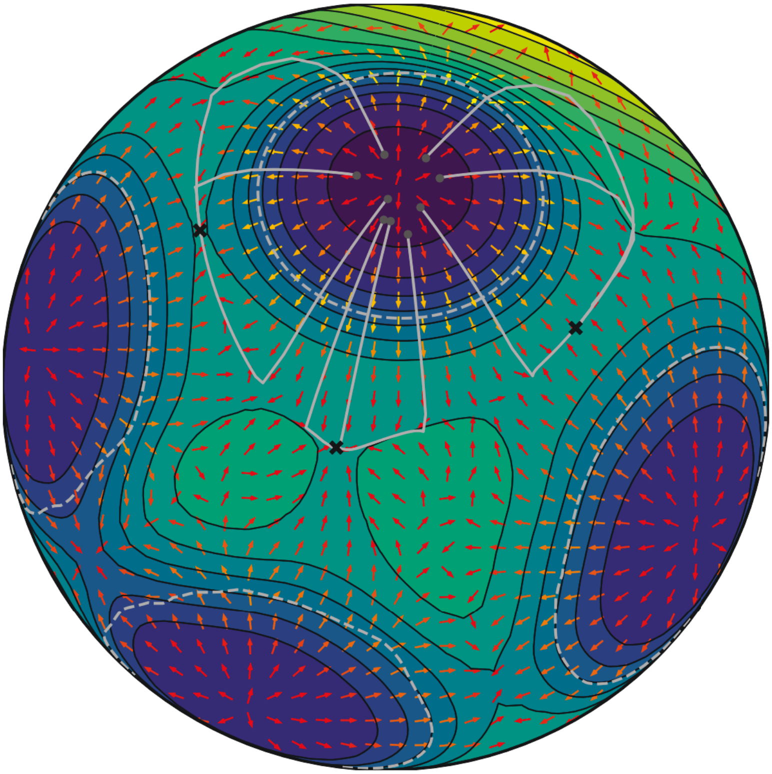


FIGURE D.3 – Spherical plot of the mode following force field in the MMF method

A single spin in a superposition of Gaussian potentials (see also Section 1.2, Chapter 2 and Appendix Section B). As the configuration space manifold $\mathcal{M}$ is a sphere (see Section I.2), the energy landscapecan be plotted, as well as the MMF search paths and force fields. The effective force of the MMF method is shown. From the gradient force field and the minimum mode orientations outside the convex region it is obvious how this force field is created and that it should point the system towards the saddle points. Note that the force field inside the convex region is simply the energy gradient, which was used in this case to escape the convex region.

E CONTRIBUTORS TO SPIRIT

The following, with minor changes, corresponds to the list of contributions to *Spirit* at time of writing this work. The approximate time frames of collaborations and contributions are noted and affiliations during this time frame are indicated. Note that the author of this work organised and coordinated the *Spirit* software project, which was initiated by Nikolai Kiselev.

Gideon Müller

- RWTH Aachen
- University of Iceland
- PGI-1/IAS-1 at Forschungszentrum Jülich

General code design and project setup (including CMake). Implementation of the core library and user interfaces, most notably: - GNEB and MMF methods - Velocity projection solver - CUDA and OpenMP parallelizations of backend - C API and Python bindings - C++ QT GUI and initial OpenGL code - Unit tests and continuous integration
(Oct. 2014 - ongoing)

Daniel Schürhoff

- RWTH Aachen
- PGI-1/IAS-1 at Forschungszentrum Jülich

Implementation of the initial core library, notably translating from Fortran90 to C++ and addition of STT to the SIB solver. Work on QT GUI and Python bindings.
(Oct. 2015 - Sept. 2016)

Nikolai Kiselev

- PGI-1/IAS-1 at Forschungszentrum Jülich

Scientific advice, general help and feedback, initial (Fortran90) implementations of: - isotropic Heisenberg Hamiltonian - Neighbour calculations - SIB solver - Monte Carlo methods
(2007 - ongoing)

Florian Rhiem

- Scientific IT-Systems, PGI/JCNS at Forschungszentrum Jülich

Implementation of C++ OpenGL code (VFRendering library), as well as JavaScript Web UI and WebGL code. Code design improvements, including the C API and CMake.
(Jan. 2016 - ongoing)

Moritz Sallermann

- RWTH Aachen
- PGI-1/IAS-1 at Forschungszentrum Jülich

Implementation of the dipole-dipole interaction using FFT convolutions.
(Apr. 2015 - Sept. 2016)

Stefanos Mavros

- RWTH Aachen
- PGI-1/IAS-1 at Forschungszentrum Jülich

Work on unit testing and documentation, implementation of the Depondt solver. Also some general code design and IO improvements.
(Apr. 2017 - Oct. 2018)

Constantin Disselkamp

- RWTH Aachen
- PGI-1/IAS-1 at Forschungszentrum Jülich

Implementation and testing of gradient approximation of spin transfer torque.
(Apr. 2017 - Jul. 2017)

Markus Hoffmann

- RWTH Aachen
- PGI-1/IAS-1 at Forschungszentrum Jülich

Bug-reports, feedback on code features and general help designing some of the functionality, user interface and input file format.
(Jun. 2016 - ongoing)

Pavel Bessarab

- Various Universities

Help with the initial GNEB implementation. Initial (Fortran90) implementation of the HTST method.
(Apr. 2015 - ongoing)

Filipp Rybakov

- Various Universities

Designs and ideas for the user interface and other code features, such as isosurfaces and colormaps for 3D systems. Some help and ideas related to code performance and CUDA.
(Jan. 2016 - ongoing)

Ingo Heimbach

- Scientific IT-Systems, PGI/JCNS at Forschungszentrum Jülich

Implementation of the initial OpenGL code. Code design suggestions and other general help.
(Jan. 2016 - ongoing)

Mathias Redies, Maximilian Merte, Rene Suckert

- RWTH Aachen

Initial CUDA implementation and tests. Code optimizations, suggestions and feedback.
(Sept. 2016 - Dec. 2016)

David Bauer

- RWTH Aachen
- PGI-1/IAS-1 at Forschungszentrum Jülich

Initial (Fortran90) implementations of the isotropic Heisenberg Hamiltonian, Neighbour calculations and the SIB solver.
(Oct. 2007 - Sept. 2008)

F USED SOFTWARE PACKAGES

Images of spin structures (arrows and isosurfaces) were created in *Spirit* [48]. Most plots and diagrams of this thesis were created using Tikz/PGF [229] with some gnuplot [230]. A few images were generated using matplotlib [231].

BIBLIOGRAPHY

[1] M. N. Baibich, J. M. Broto, A. Fert, F. N. Van Dau, F. Petroff, P. Etienne, G. Creuzet, A. Friederich, and J. Chazelas. Giant Magnetoresistance of (001)Fe/(001)Cr Magnetic Superlattices. *Physical Review Letters*, 61(21):2472–2475, Nov. 1988. doi:10.1103/PhysRevLett.61.2472.

[2] G. Binasch, P. Grünberg, F. Saurenbach, and W. Zinn. Enhanced magnetoresistance in layered magnetic structures with antiferromagnetic interlayer exchange. *Physical Review B*, 39(7):4828–4830, Mar. 1989. doi:10.1103/PhysRevB.39.4828.

[3] The Nobel Prize in Physics 2007. URL https://www.nobelprize.org/prizes/physics/2007/summary/.

[4] S. S. P. Parkin, M. Hayashi, and L. Thomas. Magnetic Domain-Wall Racetrack Memory. *Science*, 320(5873):190–194, Apr. 2008. doi:10.1126/science.1145799.

[5] S. Parkin and S.-H. Yang. Memory on the racetrack. *Nature Nanotechnology*, 10:195–198, Mar. 2015. doi:10.1038/nnano.2015.41.

[6] The Nobel Prize in Physics 2016. URL https://www.nobelprize.org/prizes/physics/2016/advanced-information/.

[7] A. N. Bogdanov and D. Yablonskii. Thermodynamically stable "vortices" in magnetically ordered crystals. The mixed state of magnets. *Zh. Eksp. Teor. Fiz*, 95(1):178, 1989. URL http://www.jetp.ac.ru/cgi-bin/e/index/e/68/1/p101?a=list.

[8] A. Bogdanov and A. Hubert. The Properties of Isolated Magnetic Vortices. *physica status solidi (b)*, 186(2):527–543, Dec. 1994. doi:10.1002/pssb.2221860223.

[9] A. Bogdanov and A. Hubert. The stability of vortex-like structures in uniaxial ferromagnets. *Journal of Magnetism and Magnetic Materials*, 195(1):182–192, Apr. 1999. doi:10.1016/S0304-8853(98)01038-5.

[10] U. K. Rößler, A. N. Bogdanov, and C. Pfleiderer. Spontaneous skyrmion ground states in magnetic metals. *Nature*, 442(7104):797–801, Aug. 2006. doi:10.1038/nature05056.

[11] U. K. Rößler, A. A. Leonov, and A. N. Bogdanov. Chiral Skyrmionic matter in non-centrosymmetric magnets. *Journal of Physics: Conference Series*, 303(1):012105, 2011. doi:10.1088/1742-6596/303/1/012105.

[12] N. S. Kiselev, A. N. Bogdanov, R. Schäfer, and U. K. Rößler. Chiral skyrmions in thin magnetic films: New objects for magnetic storage technologies? *Journal of Physics D: Applied Physics*, 44(39):392001, 2011. doi:10.1088/0022-3727/44/39/392001.

[13] F. N. Rybakov, A. B. Borisov, and A. N. Bogdanov. Three-dimensional skyrmion states in thin films of cubic helimagnets. *Physical Review B*, 87(9):094424, Mar. 2013. doi:10.1103/PhysRevB.87.094424.

[14] M. N. Wilson, A. B. Butenko, A. N. Bogdanov, and T. L. Monchesky. Chiral skyrmions in cubic helimagnet films: The role of uniaxial anisotropy. *Physical Review B*, 89(9):094411, Mar. 2014. doi:10.1103/PhysRevB.89.094411.

[15] I. Dzyaloshinsky. A thermodynamic theory of “weak” ferromagnetism of antiferromagnetics. *Journal of Physics and Chemistry of Solids*, 4(4):241–255, Jan. 1958. doi:10.1016/0022-3697(58)90076-3.

[16] T. Moriya. Anisotropic Superexchange Interaction and Weak Ferromagnetism. *Physical Review*, 120(1):91–98, Oct. 1960. doi:10.1103/PhysRev.120.91.

[17] 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(7300):901–904, June 2010. doi:10.1038/nature09124.

[18] F. Büttner, C. Moutafis, M. Schneider, B. Krüger, C. M. Günther, J. Geilhufe, C. v. K. Schmising, J. Mohanty, B. Pfau, S. Schaffert, A. Bisig, M. Foerster, T. Schulz, C. a. F. Vaz, J. H. Franken, H. J. M. Swagten, M. Kläui, and S. Eisebitt. Dynamics and inertia of skyrmionic spin structures. *Nature Physics*, 11(3):225–228, Mar. 2015. doi:10.1038/nphys3234.

[19] W. Jiang, P. Upadhyaya, W. Zhang, G. Yu, M. B. Jungfleisch, F. Y. Fradin, J. E. Pearson, Y. Tserkovnyak, K. L. Wang, O. Heinonen, S. G. E. te Velthuis, and A. Hoffmann. Blowing magnetic skyrmion bubbles. *Science*, 349(6245):283–286, July 2015. doi:10.1126/science.aaa1442.

[20] H. Du, R. Che, L. Kong, X. Zhao, C. Jin, C. Wang, J. Yang, W. Ning, R. Li, C. Jin, X. Chen, J. Zang, Y. Zhang, and M. Tian. Edge-mediated skyrmion chain and its collective dynamics in a confined geometry. *Nature Communications*, 6:8504, Oct. 2015. doi:10.1038/ncomms9504.

[21] J. Iwasaki, M. Mochizuki, and N. Nagaosa. Current-induced skyrmion dynamics in constricted geometries. *Nature Nanotechnology*, 8(10):742–747, Oct. 2013. doi:10.1038/nnano.2013.176.

[22] S. Woo, K. Litzius, B. Krüger, M.-Y. Im, L. Caretta, K. Richter, M. Mann, A. Krone, R. M. Reeve, M. Weigand, P. Agrawal, I. Lemesh, M.-A. Mawass, P. Fischer, M. Kläui, and G. S. D. Beach. Observation of room-temperature magnetic skyrmions and their current-driven dynamics in ultrathin metallic ferromagnets. *Nature Materials*, 15(5):501–506, May 2016. doi:10.1038/nmat4593.

[23] K. Litzius, I. Lemesh, 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. *Nature Physics*, 13(2):170–175, Feb. 2017. doi:10.1038/nphys4000.

[24] W. Jiang, X. Zhang, G. Yu, W. Zhang, X. Wang, M. Benjamin Jungfleisch, J. E. Pearson, X. Cheng, O. Heinonen, K. L. Wang, Y. Zhou, A. Hoffmann, and S. G. E. te Velthuis. Direct observation of the skyrmion Hall effect. *Nature Physics*, 13(2):162–169, Feb. 2017. doi:10.1038/nphys3883.

[25] 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 Nanotechnology*, 8(10):723–728, Oct. 2013. doi:10.1038/nnano.2013.174.

[26] N. Romming, C. Hanneken, M. Menzel, J. E. Bickel, B. Wolter, K. von Bergmann, A. Kubetzka, and R. Wiesendanger. Writing and Deleting Single Magnetic Skyrmions. *Science*, 341(6146):636–639, Aug. 2013. doi:10.1126/science.1240573.

[27] A. Fert, V. Cros, and J. a. Sampaio. Skyrmions on the track. *Nature Nanotechnology*, 8:152–156, Mar. 2013. doi:10.1038/nnano.2013.29.

[28] J. Sampaio, V. Cros, S. Rohart, A. Thiaville, and A. Fert. Nucleation, stability and current-induced motion of isolated magnetic skyrmions in nanostructures. *Nature Nanotechnology*, 8(11):839–844, Nov. 2013. doi:10.1038/nnano.2013.210.

[29] X. Zhang, G. P. Zhao, H. Fangohr, J. P. Liu, W. X. Xia, J. Xia, and F. J. Morvan. Skyrmion-skyrmion and skyrmion-edge repulsions in skyrmion-based racetrack memory. *Scientific Reports*, 5:7643, Jan. 2015. doi:10.1038/srep07643.

[30] J. Iwasaki, M. Mochizuki, and N. Nagaosa. Universal current-velocity relation of skyrmion motion in chiral magnets. *Nature Communications*, 4:1463, Feb. 2013. doi:10.1038/ncomms2442.

[31] I. L. Fernandes, J. Bouaziz, S. Blügel, and S. Lounis. Universality of defect-skyrmion interaction profiles. *Nature Communications*, 9(1):4395, Oct. 2018. doi:10.1038/s41467-018-06827-5.

[32] A. Thiaville, Y. Nakatani, J. Miltat, and N. Vernier. Domain wall motion by spin-polarized current: A micromagnetic study. *Journal of Applied Physics*, 95(11):7049–7051, May 2004. doi:10.1063/1.1667804.

[33] F. Büttner, I. Lemesh, and G. S. D. Beach. Theory of isolated magnetic skyrmions: From fundamentals to room temperature applications. *Scientific Reports*, 8(1):4464, Mar. 2018. doi:10.1038/s41598-018-22242-8.

[34] A. Bernand-Mantel, L. Camosi, A. Wartelle, N. Rougemaille, M. Darques, and L. Ranno. The skyrmion-bubble transition in a ferromagnetic thin film. *SciPost Physics*, 4(5):027, May 2018. doi:10.21468/SciPostPhys.4.5.027.

[35] A. Bergman, B. Skubic, J. Hellsvik, L. Nordström, A. Delin, and O. Eriksson. Ultrafast switching in a synthetic antiferromagnetic magnetic random-access memory device. *Physical Review B*, 83(22):224429, June 2011. doi:10.1103/PhysRevB.83.224429.

[36] K. V. Raman, A. M. Kamerbeek, A. Mukherjee, N. Atodiresei, T. K. Sen, P. Lazić, V. Caciuc, R. Michel, D. Stalke, S. K. Mandal, S. Blügel, M. Münzenberg, and J. S. Moodera. Interface-engineered templates for molecular spin memory devices. *Nature*, 493(7433):509–513, Jan. 2013. doi:10.1038/nature11719.

[37] S. Borlenghi, W. Wang, H. Fangohr, L. Bergqvist, and A. Delin. Designing a Spin-Seebeck Diode. *Physical Review Letters*, 112(4):047203, Jan. 2014. doi:10.1103/PhysRevLett.112.047203.

[38] P. F. Bessarab, G. P. Müller, I. S. Lobanov, F. N. Rybakov, N. S. Kiselev, H. Jónsson, V. M. Uzdin, S. Blügel, L. Bergqvist, and A. Delin. Lifetime of racetrack skyrmions. *Scientific Reports*, 8(1):3433, Feb. 2018. doi:10.1038/s41598-018-21623-3.

[39] M. Hoffmann, B. Zimmermann, G. P. Müller, D. Schürhoff, N. S. Kiselev, C. Melcher, and S. Blügel. Antiskyrmions stabilized at interfaces by anisotropic Dzyaloshinskii-Moriya interactions. *Nature Communications*, 8(1):308, Aug. 2017. doi:10.1038/s41467-017-00313-0.

[40] S. von Malottki, P. F. Bessarab, S. Haldar, A. Delin, and S. Heinze. Skyrmion lifetime in ultrathin films. *Physical Review B*, 99(6):060409, Feb. 2019. doi:10.1103/PhysRevB.99.060409.

[41] I. Žutić, J. Fabian, and S. Das Sarma. Spintronics: Fundamentals and applications. *Reviews of Modern Physics*, 76(2):323–410, Apr. 2004. doi:10.1103/RevModPhys.76.323.

[42] S. D. Bader. Colloquium: Opportunities in nanomagnetism. *Reviews of Modern Physics*, 78(1):1–15, Jan. 2006. doi:10.1103/RevModPhys.78.1.

[43] A. B. Borisov and F. N. Rybakov. Three-dimensional magnetic solitons. *The Physics of Metals and Metallography*, 112(7):745–766, Dec. 2011. doi:10.1134/S0031918X11070040.

[44] P. Sutcliffe. Skyrmion Knots in Frustrated Magnets. *Physical Review Letters*, 118(24):247203, June 2017. doi:10.1103/PhysRevLett.118.247203.

[45] P. Sutcliffe. Hopfions in chiral magnets. *Journal of Physics A: Mathematical and Theoretical*, 51(37):375401, 2018. doi:10.1088/1751-8121/aad521.

[46] Y. Liu, R. Lake, and J. Zang. Binding a Hopfion in Chiral Magnet Nanodisk. *arXiv:1806.01682 [cond-mat]*, June 2018, 1806.01682. URL http://arxiv.org/abs/1806.01682.

[47] G. P. Müller, M. Hoffmann, C. Disselkamp, D. Schürhoff, S. Mavros, M. Sallermann, N. S. Kiselev, H. Jónsson, and S. Blügel. Spirit: Multifunctional Framework for Atomistic Spin Simulations. *arXiv:1901.11350 [cond-mat]*, Jan. 2019, 1901.11350. URL http://arxiv.org/abs/1901.11350.

[48] Spirit - Spin Simulation Framework. URL https://spirit-code.github.io.

[49] A. Vansteenkiste, J. Leliaert, M. Dvornik, M. Helsen, F. Garcia-Sanchez, and B. Van Waeyenberge. The design and verification of MuMax3. *AIP Advances*, 4(10):107133, Oct. 2014. doi:10.1063/1.4899186.

[50] OOMMF. URL https://math.nist.gov/oommf/.

[51] 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. doi:10.1088/0953-8984/20/31/315203.

[52] R. F. L. Evans, W. J. Fan, P. Chureemart, T. A. Ostler, M. O. A. Ellis, and R. W. Chantrell. Atomistic spin model simulations of magnetic nanomaterials. *Journal of Physics: Condensed Matter*, 26(10):103202, 2014. doi:10.1088/0953-8984/26/10/103202.

[53] D. Bauer. *Atomistic Spin-Dynamics in Confined Magnetic Nano-Structures*. Diploma Thesis, RWTH Aachen, 2008. URL http://www.fz-juelich.de/SharedDocs/Downloads/PGI/PGI-1/EN/Bauer.diploma_pdf.pdf;jsessionid=29B54EB4F5BF643A4A7CA8B702B79C4C?__blob=publicationFile.

[54] D. S. G. Bauer, P. Mavropoulos, S. Lounis, and S. Blügel. Thermally activated magnetization reversal in monatomic magnetic chains on surfaces studied by

classical atomistic spin-dynamics simulations. *Journal of Physics: Condensed Matter*, 23(39):394204, 2011. doi:10.1088/0953-8984/23/39/394204.

[55] U. Nowak. *Thermally Activated Reversal in Magnetic Nanostructures*, volume IX of *Annual Reviews of Computational Physics*. Singapore: World Scientific, p 105, ed d stauffer edition, 2001.

[56] P. F. Bessarab, V. M. Uzdin, and H. Jónsson. Method for finding mechanism and activation energy of magnetic transitions, applied to skyrmion and antivortex annihilation. *Computer Physics Communications*, 196:335–347, Nov. 2015. doi:10.1016/j.cpc.2015.07.001.

[57] P. F. Bessarab, V. M. Uzdin, and H. Jónsson. Harmonic transition-state theory of thermal spin transitions. *Physical Review B*, 85(18):184409, May 2012. doi:10.1103/PhysRevB.85.184409.

[58] G. P. Müller, P. F. Bessarab, S. M. Vlasov, F. Lux, N. S. Kiselev, S. Blügel, V. M. Uzdin, and H. Jónsson. Duplication, Collapse, and Escape of Magnetic Skyrmions Revealed Using a Systematic Saddle Point Search Method. *Physical Review Letters*, 121(19):197202, Nov. 2018. doi:10.1103/PhysRevLett.121.197202.

[59] H.-B. Braun. Topological effects in nanomagnetism: From superparamagnetism to chiral quantum solitons. *Advances in Physics*, 61(1):1–116, Feb. 2012. doi:10.1080/00018732.2012.663070.

[60] F. Rhiem. VFRendering: A vector field rendering library, Oct. 2018. URL `https://github.com/FlorianRhiem/VFRendering`.

[61] G. P. Müller, F. N. Rybakov, H. Jónsson, S. Blügel, and N. S. Kiselev. Coupled quasimonopoles in chiral magnets. *arXiv:1903.12446 [cond-mat]*, Mar. 2019, 1903.12446. URL `http://arxiv.org/abs/1903.12446`.

[62] C. Etz, L. Bergqvist, A. Bergman, A. Taroni, and O. Eriksson. Atomistic spin dynamics and surface magnons. *Journal of Physics: Condensed Matter*, 27(24):243202, 2015. doi:10.1088/0953-8984/27/24/243202.

[63] O. Eriksson, A. Bergman, L. Bergqvist, and J. Hellsvik. *Atomistic Spin Dynamics: Foundations and Applications*. Oxford University Press, Oxford, New York, Feb. 2017.

[64] M. V. You and V. Heine. Magnetism in transition metals at finite temperatures. I. Computational model. *Journal of Physics F: Metal Physics*, 12(1):177, 1982. doi:10.1088/0305-4608/12/1/016.

[65] J. Staunton, B. L. Gyorffy, A. J. Pindor, G. M. Stocks, and H. Winter. Electronic structure of metallic ferromagnets above the Curie temperature. *Journal of Physics F: Metal Physics*, 15(6):1387, 1985. doi:10.1088/0305-4608/15/6/019.

[66] V. P. Antropov, M. I. Katsnelson, B. N. Harmon, M. van Schilfgaarde, and D. Kusnezov. Spin dynamics in magnets: Equation of motion and finite temperature effects. *Physical Review B*, 54(2):1019–1035, July 1996. doi:10.1103/PhysRevB.54.1019.

[67] S. V. Halilov, H. Eschrig, A. Y. Perlov, and P. M. Oppeneer. Adiabatic spin dynamics from spin-density-functional theory: Application to Fe, Co, and Ni. *Physical Review B*, 58(1):293–302, July 1998. doi:10.1103/PhysRevB.58.293.

[68] R. Gebauer and S. Baroni. Magnons in real materials from density-functional theory. *Physical Review B*, 61(10):R6459–R6462, Mar. 2000. doi:10.1103/PhysRevB.61.R6459.

[69] A. Aharoni and R. K. P. o. T. M. A. Aharoni. *Introduction to the Theory of Ferromagnetism*. Oxford University Press, 2000.

[70] G. T. Rado. *Magnetism: A Treatise on Modern Theory and Materials. 3. Spin Arrangements and Crystal Structure, Domains, and Micromagnetics*. Academic Press, 1963.

[71] S. Blügel, Y. Mokrousov, T. Schäpers, and Y. Ando. *Topological Matter - Topological Insulators, Skyrmions and Majoranas*. Number 139 in Schriften Des Forschungszentrums Jülich Reihe Schlüsseltechnologien / Key Technologies. Forschungszentrum Jülich GmbH Zentralbibliothek, Verlag, Jülich, 2017.

[72] M. Hoffmann and S. Blügel. Systematic derivation of realistic spin-models for beyond-Heisenberg solids from microscopic model. *arXiv:1803.01315 [cond-mat]*, Mar. 2018, 1803.01315. URL `http://arxiv.org/abs/1803.01315`.

[73] A. Szilva, M. Costa, A. Bergman, L. Szunyogh, L. Nordström, and O. Eriksson. Interatomic Exchange Interactions for Finite-Temperature Magnetism and Nonequilibrium Spin Dynamics. *Physical Review Letters*, 111(12):127204, Sept. 2013. doi:10.1103/PhysRevLett.111.127204.

[74] A. Krönlein, M. Schmitt, M. Hoffmann, J. Kemmer, N. Seubert, M. Vogt, J. Küspert, M. Böhme, B. Alonazi, J. Kügel, H. A. Albrithen, M. Bode, G. Bihlmayer, and S. Blügel. Magnetic Ground State Stabilized by Three-Site Interactions: $\mathrm{Fe}/\mathrm{Rh}(111)$. *Physical Review Letters*, 120(20):207202, May 2018. doi:10.1103/PhysRevLett.120.207202.

[75] S. Heinze, K. von Bergmann, M. Menzel, J. Brede, A. Kubetzka, R. Wiesendanger, G. Bihlmayer, and S. Blügel. Spontaneous atomic-scale magnetic skyrmion lattice in two dimensions. *Nature Physics*, 7(9):713–718, Sept. 2011. doi:10.1038/nphys2045.

[76] S. Alexander and P. W. Anderson. Interaction Between Localized States in Metals. *Physical Review*, 133(6A):A1594–A1603, Mar. 1964. doi:10.1103/PhysRev.133.A1594.

[77] P. F. Bessarab, V. M. Uzdin, and H. Jónsson. Calculations of magnetic states and minimum energy paths of transitions using a noncollinear extension of the Alexander-Anderson model and a magnetic force theorem. *Physical Review B*, 89(21):214424, June 2014. doi:10.1103/PhysRevB.89.214424.

[78] L. D. Landau and E. Lifshitz. On the theory of the dispersion of magnetic permeability in ferromagnetic bodies. *Physikalische Zeitschrift der Sowjetunion*, 8:153, 1935.

[79] W. Brown Jr. Magnetostatic Principles in Ferromagnetism. Series of Monographs on Selected Topicss in Solid State Physics. 1962.

[80] W. F. Brown. *Micromagnetics*. Number 18. Interscience Publishers, 1963.

[81] L. Néel. L'anisotropie superficielle des substances ferromagnetiques. *Comptes Rendus Hebdomadaires Des Seances De L Academie Des Sciences*, 237(23):1468–1470, 1953.

[82] C. Melcher. Chiral skyrmions in the plane. *Proc. R. Soc. A*, 470(2172):20140394, Dec. 2014. doi:10.1098/rspa.2014.0394.

[83] V. Bar'yahtar. VG Bar'yahtar and EP Stefanovski, Fiz. Tverd. Tela 11, 1946 (1969). *Fiz. Tverd. Tela*, 11:1946, 1969.

[84] P. Bak and M. H. Jensen. Theory of helical magnetic structures and phase transitions in MnSi and FeGe. *Journal of Physics C: Solid State Physics*, 13(31):L881, 1980. doi:10.1088/0022-3719/13/31/002.

[85] M. Hinzen. *Magnetisierungsfelder in Ferromagnetischen Kristallen Mit Dzyaloshinskii-Moriya-Wechselwirkung*. Master Thesis, RWTH Aachen, 2015.

[86] J. Müller. *Magnetic Skyrmions and Topological Domain Walls*. PhD Thesis, Universität zu Köln, 2018.

[87] M. Nakahara. *Geometry, Topology and Physics*. Taylor and Francis, 2003.

[88] J. M. Lee. *Riemannian Manifolds: An Introduction to Curvature*, volume 176. Springer Science & Business Media, 2006.

[89] P.-A. Absil, R. Mahony, and J. Trumpf. An Extrinsic Look at the Riemannian Hessian. In *Geometric Science of Information*, Lecture Notes in Computer Science, pages 361–368. Springer, Berlin, Heidelberg, 2013. doi:10.1007/978-3-642-40020-9_39.

[90] N. Metropolis, A. W. Rosenbluth, M. N. Rosenbluth, A. H. Teller, and E. Teller. Equation of State Calculations by Fast Computing Machines. *The Journal of Chemical Physics*, 21(6):1087–1092, June 1953. doi:10.1063/1.1699114.

[91] D. P. Landau and K. Binder. A Guide to Monte Carlo Simulations in Statistical Physics, Sept. 2009. doi:10.1017/CBO9780511994944.

[92] M. E. J. Newman and G. T. Barkema. *Monte Carlo Methods in Statistical Physics*. Oxford University Press, Oxford, New York, Feb. 1999.

[93] D. Hinzke and U. Nowak. Monte Carlo simulation of magnetization switching in a Heisenberg model for small ferromagnetic particles. *Computer Physics Communications*, 121-122:334–337, Sept. 1999. doi:10.1016/S0010-4655(99)00348-3.

[94] K. Binder and D. W. Heermann. Theoretical Foundations of the Monte Carlo Method and Its Applications in Statistical Physics. In K. Binder and D. W. Heermann, editors, *Monte Carlo Simulation in Statistical Physics: An Introduction*, Graduate Texts in Physics, pages 5–67. Springer Berlin Heidelberg, Berlin, Heidelberg, 2010. doi:10.1007/978-3-642-03163-2_2.

[95] P. W. Anderson. Theory of Magnetic Exchange Interactions:Exchange in Insulators and Semiconductors. In F. Seitz and D. Turnbull, editors, *Solid State Physics*, volume 14, pages 99–214. Academic Press, Jan. 1963. doi:10.1016/S0081-1947(08)60260-X.

[96] L. S. Campana, A. C. D'Auria, M. D'Ambrosio, U. Esposito, L. De Cesare, and G. Kamieniarz. Spectral-density method for classical systems: Heisenberg ferromagnet. *Physical Review B*, 30(5):2769–2775, Sept. 1984. doi:10.1103/PhysRevB.30.2769.

[97] G. A. Baker, H. E. Gilbert, J. Eve, and G. S. Rushbrooke. High-Temperature Expansions for the Spin-\textonehalf{} Heisenberg Model. *Physical Review*, 164(2):800–817, Dec. 1967. doi:10.1103/PhysRev.164.800.

[98] Y. Rocio. Modeling of macroscopic anisotropies due to surface effects in magnetic thin films and nanoparticles, 2011. URL `http://www.fgarciasanchez.es/thesisrocio/index.html`.

[99] K. Binder. Finite size scaling analysis of ising model block distribution functions. *Zeitschrift für Physik B Condensed Matter*, 43(2):119–140, June 1981. doi:10.1007/BF01293604.

[100] G. Marsaglia and W. W. Tsang. The Ziggurat Method for Generating Random Variables | Marsaglia | Journal of Statistical Software. *Journal of Statistical Software*, 5(8), 2000. doi:10.18637/jss.v005.i08.

[101] R. H. Swendsen and J.-S. Wang. Replica Monte Carlo Simulation of Spin-Glasses. *Physical Review Letters*, 57(21):2607–2609, Nov. 1986. doi:10.1103/PhysRevLett.57.2607.

[102] K. Hukushima and K. Nemoto. Exchange Monte Carlo Method and Application to Spin Glass Simulations. *Journal of the Physical Society of Japan*, 65(6):1604–1608, June 1996. doi:10.1143/JPSJ.65.1604.

[103] M. Böttcher, S. Heinze, S. Egorov, J. Sinova, and B. Dupé. B-T phase diagram of Pd/Fe/Ir(111) computed with parallel tempering Monte Carlo. *arXiv:1707.01708 [cond-mat]*, July 2017, 1707.01708. URL `http://arxiv.org/abs/1707.01708`.

[104] V. P. Antropov, M. I. Katsnelson, M. van Schilfgaarde, and B. N. Harmon. Ab Initio Spin Dynamics in Magnets. *Physical Review Letters*, 75(4):729–732, July 1995. doi:10.1103/PhysRevLett.75.729.

[105] T. L. Gilbert. A phenomenological theory of damping in ferromagnetic materials. *IEEE Transactions on Magnetics*, 40(6):3443–3449, Nov. 2004. doi:10.1109/TMAG.2004.836740.

[106] W. F. Brown. Thermal Fluctuations of a Single-Domain Particle. *Physical Review*, 130(5):1677–1686, June 1963. doi:10.1103/PhysRev.130.1677.

[107] C. Schieback, M. Kläui, U. Nowak, U. Rüdiger, and P. Nielaba. Numerical investigation of spin-torque using the Heisenberg model. *The European Physical Journal B*, 59(4):429–433, Oct. 2007. doi:10.1140/epjb/e2007-00062-2.

[108] G. P. Müller. *Exploration of Skyrmion Energy Landscapes*. Master Thesis, RWTH Aachen, 2015. URL `http://www.fz-juelich.de/SharedDocs/Downloads/PGI/PGI-1/EN/Mueller-Gideon-Master.pdf;jsessionid=29B54EB4F5BF643A4A7CA8B702B79C4C?__blob=publicationFile`.

[109] S. Zhang and Z. Li. Roles of Nonequilibrium Conduction Electrons on the Magnetization Dynamics of Ferromagnets. *Physical Review Letters*, 93(12):127204, Sept. 2004. doi:10.1103/PhysRevLett.93.127204.

[110] Z. Li and S. Zhang. Domain-wall dynamics driven by adiabatic spin-transfer torques. *Physical Review B*, 70(2):024417, July 2004. doi:10.1103/PhysRevB.70.024417.

[111] P. Chureemart, R. F. L. Evans, and R. W. Chantrell. Dynamics of domain wall driven by spin-transfer torque. *Physical Review B*, 83(18):184416, May 2011. doi:10.1103/PhysRevB.83.184416.

[112] Z. Li and S. Zhang. Domain-Wall Dynamics and Spin-Wave Excitations with Spin-Transfer Torques. *Physical Review Letters*, 92(20):207203, May 2004. doi:10.1103/PhysRevLett.92.207203.

[113] S. Zhang, P. M. Levy, and A. Fert. Mechanisms of Spin-Polarized Current-Driven Magnetization Switching. *Physical Review Letters*, 88(23):236601, May 2002. doi:10.1103/PhysRevLett.88.236601.

[114] C. Disselkamp. *Modelling In-Plane Spin Currents in Atomistic Spin-Dynamics*. Bachelor, RWTH Aachen, July 2017. URL `http://www.fz-juelich.de/SharedDocs/Downloads/PGI/PGI-1/EN/Disselkamp-Bachelor.pdf?__blob=publicationFile`.

[115] H. B. Callen and T. A. Welton. Irreversibility and Generalized Noise. *Physical Review*, 83(1):34–40, July 1951. doi:10.1103/PhysRev.83.34.

[116] R. Kubo. The fluctuation-dissipation theorem. *Reports on Progress in Physics*, 29(1):255, 1966. doi:10.1088/0034-4885/29/1/306.

[117] M. Plischke and B. Bergersen. *Equilibrium Statistical Physics —Solutions Manual*. World Scientific, 1994.

[118] J. H. Mentink, M. V. Tretyakov, A. Fasolino, M. I. Katsnelson, and T. Rasing. Stable and fast semi-implicit integration of the stochastic Landau–Lifshitz equation. *Journal of Physics: Condensed Matter*, 22(17):176001, 2010. doi:10.1088/0953-8984/22/17/176001.

[119] L. Rózsa, L. Udvardi, and L. Szunyogh. Langevin spin dynamics based on ab initio calculations: Numerical schemes and applications. *Journal of Physics: Condensed Matter*, 26(21):216003, 2014. doi:10.1088/0953-8984/26/21/216003.

[120] P. Depondt and F. G. Mertens. Spin dynamics simulations of two-dimensional clusters with Heisenberg and dipole–dipole interactions. *Journal of Physics: Condensed Matter*, 21(33):336005, 2009. doi:10.1088/0953-8984/21/33/336005.

[121] P. Babkevich, A. Poole, R. D. Johnson, B. Roessli, D. Prabhakaran, and A. T. Boothroyd. Electric field control of chiral magnetic domains in the high-temperature multiferroic CuO. *Physical Review B*, 85(13):134428, Apr. 2012. doi:10.1103/PhysRevB.85.134428.

[122] G. Brown, M. A. Novotny, and P. A. Rikvold. Langevin simulation of thermally activated magnetization reversal in nanoscale pillars. *Physical Review B*, 64(13):134422, Sept. 2001. doi:10.1103/PhysRevB.64.134422.

[123] U. Nowak, O. N. Mryasov, R. Wieser, K. Guslienko, and R. W. Chantrell. Spin dynamics of magnetic nanoparticles: Beyond Brown's theory. *Physical Review B*, 72(17):172410, Nov. 2005. doi:10.1103/PhysRevB.72.172410.

[124] S. K. Kim and Y. Tserkovnyak. Landau-Lifshitz theory of thermomagnonic torque. *Physical Review B*, 92(2):020410, July 2015. doi:10.1103/PhysRevB.92.020410.

[125] J. Chico, C. Etz, L. Bergqvist, O. Eriksson, J. Fransson, A. Delin, and A. Bergman. Thermally driven domain-wall motion in Fe on W(110). *Physical Review B*, 90(1):014434, July 2014. doi:10.1103/PhysRevB.90.014434.

[126] S. Selzer, U. Atxitia, U. Ritzmann, D. Hinzke, and U. Nowak. Inertia-Free Thermally Driven Domain-Wall Motion in Antiferromagnets. *Physical Review Letters*, 117(10):107201, Aug. 2016. doi:10.1103/PhysRevLett.117.107201.

[127] A. A. Kovalev. Skyrmionic spin Seebeck effect via dissipative thermomagnonic torques. *Physical Review B*, 89(24):241101, June 2014. doi:10.1103/PhysRevB.89.241101.

[128] A. Thiaville, Y. Nakatani, J. Miltat, and Y. Suzuki. Micromagnetic understanding of current-driven domain wall motion in patterned nanowires. *EPL (Europhysics Letters)*, 69(6):990, Feb. 2005. doi:10.1209/epl/i2004-10452-6.

[129] N. L. Schryer and L. R. Walker. The motion of 180° domain walls in uniform dc magnetic fields. *Journal of Applied Physics*, 45(12):5406–5421, Dec. 1974. doi:10.1063/1.1663252.

[130] V. M. Uzdin, M. N. Potkina, I. S. Lobanov, P. F. Bessarab, and H. Jónsson. The effect of confinement and defects on the thermal stability of skyrmions. *Physica B: Condensed Matter*, Sept. 2017. doi:10.1016/j.physb.2017.09.040.

[131] D. C. B. Schürhoff. *Atomistic Spin Dynamics With Real Time Control Of Simulation Parameters And Visualisation*. Master Thesis, RWTH Aachen, 2016. URL `http://www.fz-juelich.de/SharedDocs/Downloads/PGI/PGI-1/EN/Schuerhoff-Master-Thesis.pdf;jsessionid=29B54EB4F5BF643A4A7CA8B702B79C4C?__blob=publicationFile`.

[132] S. Y. Liashko, V. M. Uzdin, and H. Jónsson. Rate of thermal transitions in kagome spin ice. *Journal of Physics: Conference Series*, 741(1):012182, 2016. doi:10.1088/1742-6596/741/1/012182.

[133] S. Y. Liashko, H. Jónsson, and V. M. Uzdin. The effect of temperature and external field on transitions in elements of kagome spin ice. *New Journal of Physics*, 19(11):113008, 2017. doi:10.1088/1367-2630/aa8b96.

[134] S. Y. Liashko, V. M. Uzdin, and H. Jónsson. Energy surface and transition rates in a hexagonal element of spin ice. *Journal of Physics: Conference Series*, 903(1):012006, 2017. doi:10.1088/1742-6596/903/1/012006.

[135] H. Jónsson, G. Mills, and K. W. Jacobsen. Nudged elastic band method for finding minimum energy paths of transitions. In *Classical and Quantum Dynamics in Condensed Phase Simulations*, pages 385–404. WORLD SCIENTIFIC, June 1998. doi:10.1142/9789812839664_0016.

[136] F. N. Rybakov, A. B. Borisov, S. Blügel, and N. S. Kiselev. New Type of Stable Particlelike States in Chiral Magnets. *Physical Review Letters*, 115(11):117201, Sept. 2015. doi:10.1103/PhysRevLett.115.117201.

[137] G. Henkelman and H. Jónsson. Improved tangent estimate in the nudged elastic band method for finding minimum energy paths and saddle points. *The Journal of Chemical Physics*, 113(22):9978–9985, Nov. 2000. doi:10.1063/1.1323224.

[138] F. N. Rybakov and N. S. Kiselev. Chiral Magnetic Skyrmions with Arbitrary Topological Charge ("skyrmionic sacks"). *arXiv:1806.00782 [cond-mat, physics:nlin]*, June 2018, 1806.00782. URL `http://arxiv.org/abs/1806.00782`.

[139] J. Hagemeister, A. Siemens, L. Rózsa, E. Y. Vedmedenko, and R. Wiesendanger. Controlled creation and stability of $k\ensuremath{\pi}$ skyrmions on a discrete lattice. *Physical Review B*, 97(17):174436, May 2018. doi:10.1103/PhysRevB.97.174436.

[140] M. Finazzi, M. Savoini, A. R. Khorsand, A. Tsukamoto, A. Itoh, L. Duò, A. Kirilyuk, T. Rasing, and M. Ezawa. Laser-Induced Magnetic Nanostructures with Tunable Topological Properties. *Physical Review Letters*, 110(17):177205, Apr. 2013. doi:10.1103/PhysRevLett.110.177205.

[141] S. Komineas and N. Papanicolaou. Skyrmion dynamics in chiral ferromagnets under spin-transfer torque. *Physical Review B*, 92(17):174405, Nov. 2015. doi:10.1103/PhysRevB.92.174405.

[142] C. Heo, N. S. Kiselev, A. K. Nandy, S. Blügel, and T. Rasing. Switching of chiral magnetic skyrmions by picosecond magnetic field pulses via transient topological states. *Scientific Reports*, 6:27146, June 2016. doi:10.1038/srep27146.

[143] I. M. Singer and J. A. Thorpe. *Lecture Notes on Elementary Topology and Geometry*. Springer, 2015.

[144] L. Desplat, D. Suess, J.-V. Kim, and R. L. Stamps. Thermal stability of metastable magnetic skyrmions: Entropic narrowing and significance of internal eigenmodes. *Physical Review B*, 98(13):134407, Oct. 2018. doi:10.1103/PhysRevB.98.134407.

[145] M. Hoffmann, G. P. Müller, and S. Blügel. Engineering skyrmion lifetimes by layering and frustration. 2019, in preparation.

[146] P. F. Bessarab, V. M. Uzdin, and H. Jónsson. Size and Shape Dependence of Thermal Spin Transitions in Nanoislands. *Physical Review Letters*, 110(2):020604, Jan. 2013. doi:10.1103/PhysRevLett.110.020604.

[147] E. Wigner. The transition state method. *Transactions of the Faraday Society*, 34(0):29–41, Jan. 1938. doi:10.1039/TF9383400029.

[148] D. G. Truhlar, B. C. Garrett, and S. J. Klippenstein. Current Status of Transition-State Theory. *The Journal of Physical Chemistry*, 100(31):12771–12800, Jan. 1996. doi:10.1021/jp953748q.

[149] W. H. Miller. Importance of nonseparability in quantum mechanical transition-state theory. *Accounts of Chemical Research*, 9(8):306–312, Aug. 1976. doi:10.1021/ar50104a005.

[150] J. C. Keck. Advances in Chemical Physics. In *Advances in Chemical Physics*, volume 13 of *Advances in Chemical Physics*, page 85. John Wiley & Sons, Inc., Hoboken, NJ, USA, 1967. doi:10.1002/9780470508602.

[151] A. F. Voter and J. D. Doll. Dynamical corrections to transition state theory for multistate systems: Surface self-diffusion in the rare-event regime. *The Journal of Chemical Physics*, 82(1):80–92, Jan. 1985. doi:10.1063/1.448739.

[152] E. Pollak and P. Pechukas. Transition states, trapped trajectories, and classical bound states embedded in the continuum. *The Journal of Chemical Physics*, 69(3):1218–1226, Aug. 1978. doi:10.1063/1.436658.

[153] W. T. Coffey, D. A. Garanin, and D. J. Mccarthy. Crossover formulas in the kramers theory of thermally activated escape rates—application to spin systems. In *Advances in Chemical Physics*, pages 483–765. John Wiley & Sons, Ltd, 2007. doi:10.1002/9780470141779.ch5.

[154] P. F. Bessarab, V. M. Uzdin, and H. Jónsson. Potential Energy Surfaces and Rates of Spin Transitions. *Zeitschrift für Physikalische Chemie*, 227(9-11):1543–1557, 2013. doi:10.1524/zpch.2013.0403.

[155] P. F. Bessarab. *Theoretical Description of Stability and Transitions between Magnetic States*. Ph.D., University of Iceland, 2013.

[156] P. Loxley. *Nucleation of Solitons in the Presence of Defects*. PhD thesis, University of Western Australia, 2004. URL `http://research-repository.uwa.edu.au/en/publications/record(853d8ed1-7dd8-4134-be93-f1af8755415b).html`.

[157] L. Rózsa, J. Hagemeister, E. Y. Vedmedenko, and R. Wiesendanger. Localized spin waves in isolated $k\ensuremath{\pi}$ skyrmions. *Physical Review B*, 98(22):224426, Dec. 2018. doi:10.1103/PhysRevB.98.224426.

[158] L. Rózsa, J. Hagemeister, E. Y. Vedmedenko, and R. Wiesendanger. Effective damping enhancement in noncollinear spin structures. *Physical Review B*, 98(10):100404, Sept. 2018. doi:10.1103/PhysRevB.98.100404.

[159] N. Nagaosa and Y. Tokura. Topological properties and dynamics of magnetic skyrmions. *Nature Nanotechnology*, 8(12):899–911, Dec. 2013. doi:10.1038/nnano.2013.243.

[160] 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, Oct. 2014. doi:10.1038/srep06784.

[161] G. Pizzi, A. Cepellotti, R. Sabatini, N. Marzari, and B. Kozinsky. AiiDA: Automated interactive infrastructure and database for computational science. *Computational Materials Science*, 111:218–230, Jan. 2016. doi:10.1016/j.commatsci.2015.09.013.

[162] A. H. Larsen, J. J. Mortensen, J. Blomqvist, I. E. Castelli, R. Christensen, Marcin Dułak, J. Friis, M. N. Groves, B. Hammer, C. Hargus, E. D. Hermes, P. C. Jennings, P. B. Jensen, J. Kermode, J. R. Kitchin, E. L. Kolsbjerg, J. Kubal, Kristen Kaasbjerg, S. Lysgaard, J. B. Maronsson, T. Maxson, T. Olsen, L. Pastewka, Andrew Peterson, C. Rostgaard, J. Schiøtz, O. Schütt, M. Strange, K. S. Thygesen, Tejs Vegge, L. Vilhelmsen, M. Walter, Z. Zeng, and K. W. Jacobsen. The atomic simulation environment—a Python library for working with atoms. *Journal of Physics: Condensed Matter*, 29(27):273002, 2017. doi:10.1088/1361-648X/aa680e.

[163] B. Peters. *Reaction Rate Theory and Rare Events*. Elsevier, Mar. 2017.

[164] H. Jónsson. Simulation of surface processes. *Proceedings of the National Academy of Sciences*, 108(3):944–949, Jan. 2011. doi:10.1073/pnas.1006670108.

[165] G. H. Jóhannesson and H. Jónsson. Optimization of hyperplanar transition states. *The Journal of Chemical Physics*, 115(21):9644–9656, Nov. 2001. doi:10.1063/1.1415499.

[166] G. Henkelman and H. Jónsson. A dimer method for finding saddle points on high dimensional potential surfaces using only first derivatives. *The Journal of Chemical Physics*, 111(15):7010–7022, Oct. 1999. doi:10.1063/1.480097.

[167] M. Plasencia Gutiérrez, C. Argáez, and H. Jónsson. Improved Minimum Mode Following Method for Finding First Order Saddle Points. *Journal of Chemical Theory and Computation*, 13(1):125–134, Jan. 2017. doi:10.1021/acs.jctc.5b01216.

[168] T. Frankel. *The Geometry of Physics: An Introduction*. Cambridge university press, 2011.

[169] S.-Z. Lin, C. D. Batista, and A. Saxena. Internal modes of a skyrmion in the ferromagnetic state of chiral magnets. *Physical Review B*, 89(2):024415, Jan. 2014. doi:10.1103/PhysRevB.89.024415.

[170] I. S. Lobanov, H. Jónsson, and V. M. Uzdin. Mechanism and activation energy of magnetic skyrmion annihilation obtained from minimum energy path calculations. *Physical Review B*, 94(17):174418, Nov. 2016. doi:10.1103/PhysRevB.94.174418.

[171] S. von Malottki, B. Dupé, P. F. Bessarab, A. Delin, and S. Heinze. Enhanced skyrmion stability due to exchange frustration. *Scientific Reports*, 7(1):12299, Sept. 2017. doi:10.1038/s41598-017-12525-x.

[172] T. Q. Company. Qt | Cross-platform software development for embedded & desktop. URL `https://www.qt.io`.

[173] OpenMP. URL `https://www.openmp.org/`.

[174] CUDA, July 2017. URL `https://developer.nvidia.com/cuda-zone`.

[175] D. Demidov. VexCL is a C++ vector expression template library for OpenCL/CUDA: Ddemidov/vexcl, Nov. 2018. URL `https://github.com/ddemidov/vexcl`.

[176] M. Weigel. Simulating spin models on GPU. *Computer Physics Communications*, 182(9):1833–1836, Sept. 2011. doi:10.1016/j.cpc.2010.10.031.

[177] D. Hinzke and U. Nowak. Magnetization switching in nanowires: Monte Carlo study with fast Fourier transformation for dipolar fields. *Journal of Magnetism and Magnetic Materials*, 221(3):365–372, Nov. 2000. doi:10.1016/S0304-8853(00)00516-3.

[178] G. J. Bowden, G. B. G. Stenning, and G. van der Laan. Inter and intra macrocell model for point dipole–dipole energy calculations. *Journal of Physics: Condensed Matter*, 28(6):066001, 2016. doi:10.1088/0953-8984/28/6/066001.

[179] N. Hayashi, K. Saito, and Y. Nakatani. Calculation of Demagnetizing Field Distribution Based on Fast Fourier Transform of Convolution. *Japanese Journal of Applied Physics*, 35(12R):6065, Dec. 1996. doi:10.1143/JJAP.35.6065.

[180] M. Frigo and S. G. Johnson. The Design and Implementation of FFTW3. *Proceedings of the IEEE*, 93(2):216–231, 2005. Special issue on “Program Generation, Optimization, and Platform Adaptation”.

[181] cuFFT. URL https://developer.nvidia.com/cufft.

[182] A. Hubert and W. Rave. Systematic Analysis of Micromagnetic Switching Processes. *physica status solidi (b)*, 211(2):815–829, Feb. 1999. doi:10.1002/(SICI)1521-3951(199902)211:2<815::AID-PSSB815>3.0.CO;2-M.

[183] J. Kurzak and B. M. Pettitt. Massively parallel implementation of a fast multipole method for distributed memory machines. *Journal of Parallel and Distributed Computing*, 65(7):870–881, July 2005. doi:10.1016/j.jpdc.2005.02.001.

[184] E. Suli and M. David F. *An Introduction to Numerical Analysis*. Cambridge University Press, London, 2003.

[185] C. Runge. Ueber die numerische Auflösung von Differentialgleichungen. *Mathematische Annalen*, 46(2):167–178, June 1895. doi:10.1007/BF01446807.

[186] W. Kutta. Beitrag zur näherungweisen Integration totaler Differentialgleichungen. 1901.

[187] L. Verlet. Computer "Experiments" on Classical Fluids. I. Thermodynamical Properties of Lennard-Jones Molecules. *Physical Review*, 159(1):98–103, July 1967. doi:10.1103/PhysRev.159.98.

[188] W. C. Swope, H. C. Andersen, P. H. Berens, and K. R. Wilson. A computer simulation method for the calculation of equilibrium constants for the formation of physical clusters of molecules: Application to small water clusters. *The Journal of Chemical Physics*, 76(1):637–649, Jan. 1982. doi:10.1063/1.442716.

[189] J. R. Shewchuk et al. An introduction to the conjugate gradient method without the agonizing pain. 1994.

[190] M. Diehl, F. Glineur, E. Jarlebring, and W. Michiels, editors. *Recent Advances in Optimization and Its Applications in Engineering*. Springer-Verlag, Berlin Heidelberg, 2010. URL springer.com/us/book/9783642125973.

[191] P. S. Krishnaprasad and X. Tan. Cayley transforms in micromagnetics. *Physica B: Condensed Matter*, 306(1):195–199, Dec. 2001. doi:10.1016/S0921-4526(01)01003-1.

[192] D. Lewis and N. Nigam. Geometric integration on spheres and some interesting applications. *Journal of Computational and Applied Mathematics*, 151(1):141–170, Feb. 2003. doi:10.1016/S0377-0427(02)00743-4.

[193] A. V. Ivanov, V. M. Uzdin, and H. Jónsson. Fast and Robust Algorithm for the Minimisation of the Energy of Spin Systems. *arXiv:1904.02669 [physics]*, Apr. 2019, 1904.02669. URL `http://arxiv.org/abs/1904.02669`.

[194] F. Zheng, F. N. Rybakov, A. B. Borisov, D. Song, S. Wang, Z.-A. Li, H. Du, N. S. Kiselev, J. Caron, A. Kovács, M. Tian, Y. Zhang, S. Blügel, and R. E. Dunin-Borkowski. Experimental observation of chiral magnetic bobbers in B20-type FeGe. *Nature Nanotechnology*, 13(6):451–455, June 2018. doi:10.1038/s41565-018-0093-3.

[195] H. Du, X. Zhao, F. N. Rybakov, A. B. Borisov, S. Wang, J. Tang, C. Jin, C. Wang, W. Wei, N. S. Kiselev, Y. Zhang, R. Che, S. Blügel, and M. Tian. Interaction of Individual Skyrmions in a Nanostructured Cubic Chiral Magnet. *Physical Review Letters*, 120(19):197203, May 2018. doi:10.1103/PhysRevLett.120.197203.

[196] M. Redies, F. R. Lux, J.-P. Hanke, P. M. Buhl, G. P. Müller, N. S. Kiselev, S. Blügel, and Y. Mokrousov. Distinct magnetotransport and orbital fingerprints of chiral bobbers. *Physical Review B*, 99(14):140407, Apr. 2019. doi:10.1103/PhysRevB.99.140407.

[197] C. Donnelly, M. Guizar-Sicairos, V. Scagnoli, S. Gliga, M. Holler, J. Raabe, and L. J. Heyderman. Three-dimensional magnetization structures revealed with X-ray vector nanotomography. *Nature*, 547(7663):328–331, July 2017. doi:10.1038/nature23006.

[198] A. Malozemoff and J. Slonczewski. *Magnetic Domain Walls in Bubble Materials: Advances in Materials and Device Research*, volume 1. Academic press, 2016.

[199] E. Feldtkeller. Mikromagnetisch stetige und unstetige Magnetisierungskonfigurationen. *Zeitschrift fur Angewandte Physik*, 19(6):530, 1965.

[200] W. Döring. Point Singularities in Micromagnetism. *Journal of Applied Physics*, 39(2):1006–1007, Feb. 1968. doi:10.1063/1.1656144.

[201] P. Milde, D. Köhler, J. Seidel, L. M. Eng, A. Bauer, A. Chacon, J. Kindervater, S. Mühlbauer, C. Pfleiderer, S. Buhrandt, C. Schütte, and A. Rosch. Unwinding of a Skyrmion Lattice by Magnetic Monopoles. *Science*, 340(6136):1076–1080, May 2013. doi:10.1126/science.1234657.

[202] N. Kanazawa, Y. Nii, X.-X. Zhang, A. S. Mishchenko, G. De Filippis, F. Kagawa, Y. Iwasa, N. Nagaosa, and Y. Tokura. Critical phenomena of emergent magnetic monopoles in a chiral magnet. *Nature Communications*, 7:11622, May 2016. doi:10.1038/ncomms11622.

[203] G. Henkelman, B. P. Uberuaga, and H. Jónsson. A climbing image nudged elastic band method for finding saddle points and minimum energy paths. *The Journal of Chemical Physics*, 113(22):9901–9904, Nov. 2000. doi:10.1063/1.1329672.

[204] I. I. Smalyukh, Y. Lansac, N. A. Clark, and R. P. Trivedi. Three-dimensional structure and multistable optical switching of triple-twisted particle-like excitations in anisotropic fluids. *Nature Materials*, 9(2):139–145, Feb. 2010. doi:10.1038/nmat2592.

[205] A. O. Leonov and K. Inoue. Homogeneous and heterogeneous nucleation of skyrmions in thin layers of cubic helimagnets. *Physical Review B*, 98(5):054404, Aug. 2018. doi:10.1103/PhysRevB.98.054404.

[206] A. O. Leonov, T. L. Monchesky, J. C. Loudon, and A. N. Bogdanov. Three-dimensional chiral skyrmions with attractive interparticle interactions. *Journal of Physics: Condensed Matter*, 28(35):35LT01, 2016. doi:10.1088/0953-8984/28/35/35LT01.

[207] J. C. Loudon, A. O. Leonov, A. N. Bogdanov, M. C. Hatnean, and G. Balakrishnan. Direct observation of attractive skyrmions and skyrmion clusters in the cubic helimagnet ${\mathrm{Cu}}_{2}{\mathrm{OSeO}}_{3}$. *Physical Review B*, 97(13):134403, Apr. 2018. doi:10.1103/PhysRevB.97.134403.

[208] S. A. Meynell, M. N. Wilson, H. Fritzsche, A. N. Bogdanov, and T. L. Monchesky. Surface twist instabilities and skyrmion states in chiral ferromagnets. *Physical Review B*, 90(1):014406, July 2014. doi:10.1103/PhysRevB.90.014406.

[209] H. Du, D. Liang, C. Jin, L. Kong, M. J. Stolt, W. Ning, J. Yang, Y. Xing, J. Wang, R. Che, J. Zang, S. Jin, Y. Zhang, and M. Tian. Electrical probing of field-driven cascading quantized transitions of skyrmion cluster states in MnSi nanowires. *Nature Communications*, 6:7637, July 2015. doi:10.1038/ncomms8637.

[210] F. N. Rybakov, A. B. Borisov, S. Blügel, and N. S. Kiselev. New spiral state and skyrmion lattice in 3D model of chiral magnets. *New Journal of Physics*, 18(4):045002, 2016. doi:10.1088/1367-2630/18/4/045002.

[211] C. Jin, Z.-A. Li, A. Kovács, J. Caron, F. Zheng, F. N. Rybakov, N. S. Kiselev, H. Du, S. Blügel, M. Tian, Y. Zhang, M. Farle, and R. E. Dunin-Borkowski. Control of morphology and formation of highly geometrically confined magnetic skyrmions. *Nature Communications*, 8:15569, June 2017. doi:10.1038/ncomms15569.

[212] M. Charilaou and J. F. Löffler. Skyrmion oscillations in magnetic nanorods with chiral interactions. *Physical Review B*, 95(2):024409, Jan. 2017. doi:10.1103/PhysRevB.95.024409.

[213] P. A. Midgley and R. E. Dunin-Borkowski. Electron tomography and holography in materials science. *Nature Materials*, 8(4):271–280, Apr. 2009. doi:10.1038/nmat2406.

[214] S. Da Col, S. Jamet, N. Rougemaille, A. Locatelli, T. O. Mentes, B. S. Burgos, R. Afid, M. Darques, L. Cagnon, J. C. Toussaint, and O. Fruchart. Observation of Bloch-point domain walls in cylindrical magnetic nanowires. *Physical Review B*, 89(18):180405, May 2014. doi:10.1103/PhysRevB.89.180405.

[215] H. Du, J. P. DeGrave, F. Xue, D. Liang, W. Ning, J. Yang, M. Tian, Y. Zhang, and S. Jin. Highly Stable Skyrmion State in Helimagnetic MnSi Nanowires. *Nano Letters*, 14(4):2026–2032, Apr. 2014. doi:10.1021/nl5001899.

[216] L. D. Landau. *LD Landau*, volume 1. Oxford: Pergamon Press, 546, 1965.

[217] A. Abrikosov. AA Abrikosov, Sov. Phys. JETP 5, 1174 (1957). *Sov. Phys. JETP*, 5:1174, 1957.

[218] R. Chimata, A. Bergman, L. Bergqvist, B. Sanyal, and O. Eriksson. Microscopic Model for Ultrafast Remagnetization Dynamics. *Physical Review Letters*, 109(15):157201, Oct. 2012. doi:10.1103/PhysRevLett.109.157201.

[219] N. M. Rosengaard and B. Johansson. Finite-temperature study of itinerant ferromagnetism in Fe, Co, and Ni. *Physical Review B*, 55(22):14975–14986, June 1997. doi:10.1103/PhysRevB.55.14975.

[220] M. Ležaić, P. Mavropoulos, G. Bihlmayer, and S. Blügel. Exchange interactions and local-moment fluctuation corrections in ferromagnets at finite temperatures based on noncollinear density-functional calculations. *Physical Review B*, 88(13):134403, Oct. 2013. doi:10.1103/PhysRevB.88.134403.

[221] J. Jensen and A. R. Mackintosh. *Rare Earth Magnetism*. Clarendon Press Oxford, 1991.

[222] C. J. Fennie and K. M. Rabe. Magnetically Induced Phonon Anisotropy in ${\mathrm{ZnCr}}_{2}{\mathrm{O}}_{4}$ from First Principles. *Physical Review Letters*, 96(20):205505, May 2006. doi:10.1103/PhysRevLett.96.205505.

[223] M. Mochizuki, N. Furukawa, and N. Nagaosa. Spin Model of Magnetostrictions in Multiferroic Mn Perovskites. *Physical Review Letters*, 105(3):037205, July 2010. doi:10.1103/PhysRevLett.105.037205.

[224] M. Mochizuki, N. Furukawa, and N. Nagaosa. Theory of spin-phonon coupling in multiferroic manganese perovskites $R$MnO${}_{3}$. *Physical Review B*, 84(14):144409, Oct. 2011. doi:10.1103/PhysRevB.84.144409.

[225] D. Wang, J. Weerasinghe, and L. Bellaiche. Atomistic Molecular Dynamic Simulations of Multiferroics. *Physical Review Letters*, 109(6):067203, Aug. 2012. doi:10.1103/PhysRevLett.109.067203.

[226] Y. Tokura, S. Seki, and N. Nagaosa. Multiferroics of spin origin. *Reports on Progress in Physics*, 77(7):076501, 2014. doi:10.1088/0034-4885/77/7/076501.

[227] E. Mozafari, B. Alling, P. Steneteg, and I. A. Abrikosov. Role of N defects in paramagnetic CrN at finite temperatures from first principles. *Physical Review B*, 91(9):094101, Mar. 2015. doi:10.1103/PhysRevB.91.094101.

[228] W. H. Press, S. A. Teukolsky, W. T. Vetterling, and B. P. Flannery. *Numerical Recipes 3rd Edition: The Art of Scientific Computing*. Cambridge university press, 2007.

[229] PGF and TikZ. URL `https://pgf-tikz.github.io/`.

[230] T. Williams, C. Kelley, and many others. Gnuplot 4.6: An interactive plotting program. Apr. 2013. URL `http://gnuplot.sourceforge.net/`.

[231] J. D. Hunter. Matplotlib: A 2D graphics environment. *Computing In Science & Engineering*, 9(3):90–95, 2007. doi:10.1109/MCSE.2007.55.

LIST OF PUBLICATIONS

- M. Hoffmann, B. Zimmermann, **G. P. Müller**, D. Schürhoff, N. S. Kiselev, C. Melcher, S. Blügel. "Antiskyrmions stabilized at interfaces by anisotropic Dzyaloshinskii-Moriya interaction." Nature Communications **8** 308 (2017).
  – the author contributed, in equal parts with D. Schürhoff, by implementing the needed anisotropic interactions in *Spirit* and supporting simulations and the preparation of the manuscript. No figures or text have been included in this thesis.

- P. F. Bessarab, **G. P. Müller**, I. S. Lobanov, F. N. Rybakov, N. S. Kiselev, H. Jónsson, V. M. Uzdin, S. Blügel, L. Bergqvist, A. Delin. "Lifetime of racetrack skyrmions." Scientific Reports **8** 3433 (2018).
  – the author contributed to the conception and writing, as well as the discussion of simulation results. Figure 3 has been included in this thesis and parts of the text have been rewritten to suit this work.

- **G. P. Müller**, P. F. Bessarab, S. Vlasov, F. Lux, N. S. Kiselev, S. Blügel, V. M. Uzdin, H. Jónsson. "Duplication, Collapse, and Escape of Magnetic Skyrmions Revealed Using a Systematic Saddle Point Search Method." Physical Review Letters **121** 197202 (2018).
  – the author contributed to the conception of the project and derived analytical solutions for the essential equations with contributions from S. Vlasov and F. Lux. The author performed the implementation into *Spirit* and all calculations, created all figures and contributed significantly to the data analysis and all parts of the text. The text has been rewritten to suit this work and all figures have been adapted.

- M. Redies, F. Lux, P. M. Buhl, **G. P. Müller**, N. S. Kiselev, S. Blügel, Y. Mokrousov, J.-P. Hanke. "Distinct magnetotransport and orbital fingerprints of chiral bobbers." Physical Review B **99**, 140407(R) (2019). – the author contributed by providing and supporting the calculation of three-dimensional spin textures. No text or figures have been included in this thesis.

- **G. P. Müller**, M. Hoffmann, C. Dißelkamp, D. Schürhoff, S. Mavros, M. Sallermann, N. S. Kiselev, H. Jónsson, S. Blügel. "Spirit: Multifunctional Framework for Atomistic Spin Simulations." Physical Review B **99**, 224414 (2019).
  – the author has contributed significantly to the conception and writing of this article, as well as the implementation of all code, except for dipolar interactions.

The author has contributed to the calculations of section III A-C, performed the calculations of section III D-F and created all figures, except for figures 3, 6 and 12. Major parts of the text have been rewritten to suit this thesis and all figures, except figure 12 have been included.

- **G. P. Müller**, F. N. Rybakov, H. Jónsson, S. Blügel, N. S. Kiselev. "Coupled quasi-monopoles in chiral magnets." arxiv:1903.12446 (2019), under review at Physical Review Letters.
  – the author has performed the calculations and contributed significantly to the writing of the text and creation of figures. Major parts of the text have been rewritten and all figures adapted to suit this thesis.

- M. Hoffmann, **G. P. Müller**, S. Blügel. "Engineering skyrmion lifetimes by layering and frustration." In preparation.
  – the author has contributed to the conception of the study, performed the implementation in *Spirit*, supported the calculations and contributed to the writing of the text. No text or figures have been included in this thesis.

Band / Volume 192
High-resolution genome and transcriptome analysis of *Gluconobacter oxydans* 621H and growth-improved strains by next-generation sequencing
A. Kranz (2019), III, 182 pp
ISBN: 978-3-95806-385-3

Band / Volume 193
Group IV (Si)GeSn Light Emission and Lasing Studies
D. Stange (2019), vi, 151 pp
ISBN: 978-3-95806-389-1

Band / Volume 194
Construction and analysis of a spatially organized cortical network model
J. Senk (2019), 245 pp
ISBN: 978-3-95806-390-7

Band / Volume 195
Large-scale Investigations of Non-trivial Magnetic Textures in Chiral Magnets with Density Functional Theory
M. Bornemann (2019), 143 pp
ISBN: 978-3-95806-394-5

Band / Volume 196
Neutron scattering
Experimental Manuals of the JCNS Laboratory Course held at Forschungszentrum Jülich and at the Heinz-Maier-Leibnitz Zentrum Garching
edited by T. Brückel, S. Förster, G. Roth, and R. Zorn (2019),
ca 150 pp
ISBN: 978-3-95806-406-5

Band / Volume 197
Topological transport in non-Abelian spin textures from first principles
P. M. Buhl (2019), vii, 158 pp
ISBN: 978-3-95806-408-9

Band / Volume 198
Shortcut to the carbon-efficient microbial production of chemical building blocks from lignocellulose-derived D-xylose
C. Brüsseler (2019), X, 62 pp
ISBN: 978-3-95806-409-6

Band / Volume 199
Regulation and assembly of the cytochrome *bc*$_1$-*aa*$_3$ supercomplex in *Corynebacterium glutamicum*
C.-F. Davoudi (2019), 135 pp
ISBN: 978-3-95806-416-4

Band / Volume 200
Variability and compensation in Alzheimer's disease across different neuronal network scales
C. Bachmann (2019), xvi, 165 pp
ISBN: 978-3-95806-420-1

Band / Volume 201
Crystal structures and vibrational properties of chalcogenides: the role of temperature and pressure
M. G. Herrmann (2019), xi, 156 pp
ISBN: 978-3-95806-421-8

Band / Volume 202
Current-induced magnetization switching in a model epitaxial Fe/Au bilayer
P. Gospodarič (2019), vi, 120, XXXVIII pp
ISBN: 978-3-95806-423-2

Band / Volume 203
Network architecture and heme-responsive gene regulation of the two-component systems HrrSA and ChrSA
M. Keppel (2019), IV, 169 pp
ISBN: 978-3-95806-427-0

Band / Volume 204
Spin-orbitronics at the nanoscale: From analytical models to real materials
J. Bouaziz (2019), 228 pp
ISBN: 978-3-95806-429-4

Band / Volume 205
Advanced methods for atomic scale spin simulations and application to localized magnetic states
G. P. Müller (2019), xx, 194 pp
ISBN: 978-3-95806-432-4

Weitere ***Schriften des Verlags im Forschungszentrum Jülich*** unter
http://wwwzb1.fz-juelich.de/verlagextern1/index.asp

Schlüsseltechnologien / Key Technologies
Band / Volume 205
ISBN 978-3-95806-432-4

Mitglied der Helmholtz-Gemeinschaft

JÜLICH
Forschungszentrum