universität wien

MASTERARBEIT | MASTER'S THESIS

Titel | Title

Shape Optimization of Tunneling Magnetoresistance Sensor Design via Automated Micromagnetic Simulations

verfasst von | submitted by

Claas Vitus Friedrich Fillies BSc.

angestrebter akademischer Grad | in partial fulfilment of the requirements for the degree of

Master of Science (MSc)

Wien | Vienna, 2025

Studienkennzahl lt. Studienblatt | Degree programme code as it appears on the student record sheet:	UA 066 910
Studienrichtung lt. Studienblatt | Degree programme as it appears on the student record sheet:	Masterstudium Computational Science
Betreut von | Supervisor:	Dipl.-Ing. Dr.techn. Lukas Exl Privatdoz.

Abstract

Shape Optimization of Tunneling Magnetoresistance Sensor Design via Automated Micromagnetic Simulations

by Claas Fillies

This study presents an open-source, modular simulation framework for optimizing the hysteresis curves of magnetic samples. A focus is directed toward $Ni_{80}Fe_{20}$ thin-film soft magnets, intended to act as the free layers in tunneling magnetoresistance (TMR) sensors. Using Bayesian optimization a continuous size optimization is performed to compare three different shapes. The objective is to maximize the field range R_{lin}, within which the magnetizations exhibit a linear response to variations in the external magnetic field. With 0.172 T, the elliptical cylinder exhibited the broadest field range of linear response, indicating that shapes with fewer edges exhibit an enhanced R_{lin}. Analytical expressions for R_{lin} were derived, and a near-perfect linear correlation (Pearson coefficient of 0.999) to the numerical computed values is observed. Further the Bayesian optimizer is presented as a suitable tool to optimize the dimensions of the shapes, and an overview of the hyperparameter tuning is provided.
The presented simulation framework is designed in a modular fashion to enable a broad range of use cases in research and industrial applications, and guidelines for modification are given.

Acknowledgements

This work was enabled by the financial support of the European research project "Magnetic Multiscale Modelling Suite" [1], as well as the University for Continuing Education Krems. I would like to thank both institutions for their trust and support.

Further, I would like to thank Dipl.-Ing. Dr.techn. Lukas Exl, Priv.-Doz., for his excellent supervision and feedback on this master's thesis. I also thank Univ.-Doz. Dipl.-Ing. Dr. Thomas Schrefl for his guidance and oversight throughout this thesis. I am grateful to him teaching me allot both, both scientifically and personally. Particularly I want to thank Dr. Alexander Kovacs, MSc BSc, not only for countless scientific discussions and guidance, but also for his ability to draw enthusiasm from problems. Moreover, I sincerely thank Ing. Dr. Harald Özelt, MSc, Dipl.-Ing. Dr. Wilfried Hortschitz and Clemens Wager, MSc, as well as the entire research group at the DISS.

Finally, I am proud to thank my parents, Stella Hoepner-Fillies and Christian Fillies, for supporting me throughout my academic journey, as well as life in general.

Declaration

Funded by the European Union. Views and opinions expressed are however those of the author only and do not necessarily reflect those of the European Union or European Health and Digital Executive Agency (HADEA). Neither the European Union nor the granting authority can be held responsible for them.

The large-language-model ChatGPT 4.0 (OpenAI, 2024) was employed exclusively for linguistic refinement of this thesis. The scientific content was preserved without alteration.

Contents

List of Figures

List of Tables

Motivation

Magnets and the electromagnetic phenomena arising from their properties are becoming increasingly influential. From electric power generation and conversion to the magnetic sensors in smartphones, modern technology relies fundamentally on the properties of magnets [2]. Experimental research, however, is expensive and time-consuming. Therefore, micromagnetic simulations are used to explore the broad spectrum of parameter options and aid experimental methodologies in directing [3]. They play a crucial role in the development of new magnetic devices. In recent years, for example, they have enabled the development of biosensors [4], or smart living devices [5].

Scientific research groups worldwide have made significant efforts to develop more efficient and sustainable magnetic devices [6]. In 2024, the European Union reinforced its commitment to this direction by funding the Magnetic Multiscale Modelling Suite (MaMMoS) [1]. It aims to provide an open-source modeling suite for optimizing magnetic devices. The goal is to develop an integrated software framework for simulations, covering scales from the atomic level to the device level.

This work aims to support the MaMMoS research project at the device scale, focusing on optimizing the influence of device shape and size on its magnetic behavior. Particular attention will be given to the optimization of thin-film $Ni_{80}Fe_{20}$ soft magnets, which are crucial in the growing market of tunneling magnetoresistance sensors [5].

To achieve this goal, two main research fields must be connected. The first is micromagnetic simulations [6], which aim to approximate the magnetic characteristics of a given sample. The second field is Bayesian optimization [7], which is used to suggest optimized device shapes. These two fields will be integrated into a single software framework that enables automated shape and size optimization. The framework should be easy to use for various applications, require no human interaction, yet remain reliable and fully trackable. Although the usability of this framework will be demonstrated in the case of thin-film soft magnets, this should be seen as a proof of concept. The primary objective is to provide researchers and industry partners with a versatile tool for developing magnetic devices for various applications.

Therefore, this study first provides a foundational overview of micromagnetism in Chapter 1, tunneling magnetoresistance sensors as a use case in Chapter 2 and Bayesian optimization in Chapter 3. The subsequent Chapters 4–5 introduce the simulation framework and conclude with a case study on the findings for thin-film soft magnets.

Chapter 1

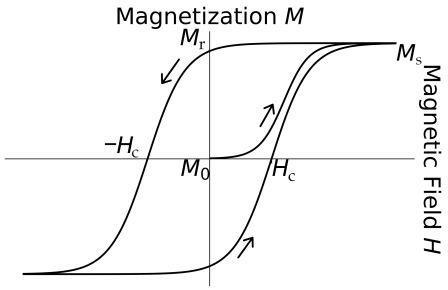
Micromagnetic Simulations

1.1 Magnetic Materials

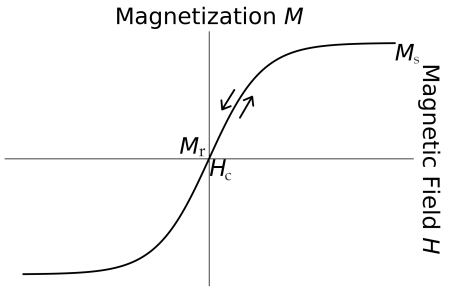
Magnetic materials can be classified by interaction of their magnetization M with an external magnetic field H. This relationship is formulated by the susceptibility χ:

$$M := \chi H \tag{1.1}$$

Depending on χ, several groups of magnetic materials can be classified. For instance, diamagnetic materials have a magnetic susceptibility of $\chi_{\text{dia}} \approx -10^{-5}$, indicating that the magnetic moments align slightly opposite to the external field. In contrast, paramagnetic materials have a small but positive $\chi_{\text{para}} \approx 10^{-3} - 10^{-5}$. Ferromagnetic materials, with $\chi_{\text{ferr}} \approx 50 - 10\,000$, are particularly important in materials science, as many of the devices discussed in the introduction rely on their properties. Since χ_{ferr} is positive, the net magnetization aligns with the external magnetic field. In practice, Eq. 1.1 can only be applied locally, because χ depends on both the field strength and the material's magnetization history. Therefore their magnetization shows a response to an external magnetic field. This behavior is described by the hysteresis loop and is depicted in Fig. 1.1. Starting from a demagnetized state M_0, an increasing external field progressively aligns more and more magnetic moments with its direction until near complete alignment is reached. This state of maximum alignment is called saturation. If there are no secondary phases and the magnet can be fully saturated, the saturation magnetization equals the spontaneous magnetization M_{s}, which is an intrinsic material property. If the external magnetic field is then reduced to zero, the magnetization also decreases along the curve of the downward-pointing arrow in Fig. 1.1. When this field is reduced to zero, the magnetization can either return to its initial state, where the sum of magnetic moments cancel each other out, or they remain aligned in the direction of the previously applied field. This residual magnetization, known as remanence M_{r}, persists even after the external field is removed. It generates an internal and external magnetostatic field. The soft magnetic sensor elements (Fig. 1.1B) have remanence close to zero, allowing them to maintain high accuracy in measuring external fields as sensors. Hard magnets (Fig. 1.1A) have remanence greater than zero, enabling their use as permanent magnets in electrical motors. If the external field is further reduced, the coercivity, H_{c}, is reached, which is the magnetic field strength required nullify the material's magnetization. Since soft magnets have low remanence, they require only a weak magnetic field to nullify their magnetization, and thus have low coercivity. Conversely, hard magnets have high coercivity. If the external field is further reduced,



(A) Hysteresis Loop of a Hard Ferromagnet ($H_c > 0$)



(B) Hysteresis Loop of a Soft Ferromagnet ($H_c \approx 0$)

FIGURE 1.1: Theoretical consideration of hysteresis loops for two ferromagnets. (A) represents a hard magnet, and (B) represents a soft magnet. The magnetization of the ferromagnets is plotted as a function of the applied magnetic field. The arrows indicate the direction of the hysteresis curve, depending on whether the magnetic field is increasing or decreasing. For hard magnets, the magnetization curve from a demagnetized state is also illustrated. Additionally, the remanence M_r, the saturation magnetization M_s, and the coercivity H_c are shown.

the point of saturation magnetization in the opposite direction is reached. If the external field is then increased, the magnetization follows the curve with the upward-pointing arrows until saturation in the initial orientation is reached again.
This is why hysteresis is called a loop: the magnetization, in response to the external field, always depends on the previous magnetization state—its history.
For soft magnetic materials, the coercive field H_c is approximately zero, ideally their hysteresis loop forms a single curve with little to no enclosed area. Hard magnets, however, with $H_c, M_r > 0$, exhibit a hysteresis loop with a significant enclosed area. Their magnetization cannot be reset to a demagnetized state by an external field.
For a detailed discussion, see Coey [8] and Barman et al. [9].

1.2 Magnetostatic State

Every stable physical state corresponds to a local or global minimum of the system's energy landscape. This principle extends to magnetostatic states as well. In magnetism, all energy terms within a material are combined into the so-called total free energy $\mathcal{E}_{tot}$. The four major contributors to the free energy are $\mathcal{E}_{ex}$, the exchange energy; $\mathcal{E}_a$, the anisotropy energy; $\mathcal{E}_Z$, the Zeeman energy; and $\mathcal{E}_{demag}$, the demagnetizing energy.
More detailed descriptions on the formulation of the total free energy and its components can be found in [10], [11] and [8].

$$\mathcal{E}_{tot} = \mathcal{E}_{ex} + \mathcal{E}_a + \mathcal{E}_Z + \mathcal{E}_{demag} \tag{1.2}$$

1.2.1 Exchange Energy

The exchange energy $\mathcal{E}_{ex}$ arises from the quantum mechanical interaction between the spins of two electrons. The Pauli exclusion principle prohibits two electrons with the same spin from occupying the same quantum state, while the Coulomb interaction causes electrons with the same spin to move apart. In ferromagnetic materials, this energy balance results in a parallel spin state. To achieve a concise notation, the unit magnetization vector $\mathbf{m}$ is introduced, which is related to the magnetization $\mathbf{M}$ through the spontaneous magnetization M_s:

$$\mathbf{m} = \frac{\mathbf{M}}{M_s} \tag{1.3}$$

$\mathbf{m}$ is a dimensionless vector that represents the local orientation of the magnetization and always has a magnitude of 1.

$$\mathcal{E}_{ex} = A \int_V (\nabla \mathbf{m})^2 dV \tag{1.4}$$

Assuming only small spatial changes in the magnetization, the exchange energy can be expressed as the integral over the magnetic sample V, as shown in (1.4), where $(\nabla \mathbf{m})^2 = (\nabla m_x)^2 + (\nabla m_y)^2 + (\nabla m_z)^2$ describes the spatial variation of the unit magnetization vector, and A is the exchange stiffness constant.

1.2.2 Anisotropy Energy

In the absence of an external magnetic field, it is still energetically favorable for the magnetic moments to align toward a common direction. These favored directions of orientation are the so-called easy axes, along which a weaker external field is required to reach saturation. In contrast to an easy axis, a higher external field is required to force the magnetization to align along the hard axis. Each sample can have multiple hard or easy axes, which do not necessarily have to be orthogonal to each other. As described in [11], material properties such as sample shape, nanocrystal structure, and atomic-scale texture, govern the orientation of the hard and easy axes. The three dominant contributions to anisotropy are crystalline anisotropy, shape anisotropy, and induced anisotropy.
Magnetocrystalline anisotropy represents the interaction of the crystallographic lattice with the magnetization. Effects like spin-orbit coupling or interatomic dipole interactions leads to directional variations in magnetization characteristics. The induced anisotropy energy results from applying external stress to a material or by annealing or depositing a material [8]. However, in $Ni_{80}Fe_{20}$ thin films, magnetocrystalline anisotropy and induced anisotropy are vanishingly small and will therefore not be further considered in this study [12]. Consequently, the configurations of the hard and easy axis in a soft magnetic thin film are governed by the shape anisotropy. Shape anisotropy describes how the sample's geometry influences the internal demagnetization field and will therefore be further discussed in Sec. 1.2.4 on demagnetizing energy.

1.2.3 Zeeman Energy

If an external magnetic field H_{ext} is applied, the magnetization $\mathbf{m}$ (unit vector) experience an interaction known as the Zeeman energy, which is minimized when the magnetization aligns with the external field. Equation 1.5 represents the Zeeman energy density integrated over the volume V. μ_0 describes the permeability in free space.

$$\mathcal{E}_{\text{Z}} = -\mu_0 M_{\text{s}} \int_V (\mathbf{m} \cdot \mathbf{H}_{\text{ext}})\, \mathrm{d}V. \tag{1.5}$$

1.2.4 Demagnetizing Energy

The Zeeman interaction favors the alignment of the magnetic moments with the external field. However, the dipole-dipole interactions among the magnetic moments can generate a field that counteracts the external field. This dipole field splits into two parts. The first component, known as the short-range interaction, primarily depends on local quantities an as the precise arrangement of the interacting magnetic moments within the lattice and its symmetry. This interaction is usually included into crystalline anisotropy and can be neglected for soft magnets [12]. The long-range interaction of the magnetic moments no longer depends on their exact position in the lattice. This interaction is described as the demagnetization field H_{demag}. The demagnetization energy results from the interaction of magnetic moments of each atom in the demagnetization field. Similar to the Zeeman energy, the demagnetization energy is expressed as an integral over the sample volume V, with a prefactor of $\frac{1}{2}$ to prevent double counting of atomic contributions [11].

$$\mathcal{E}_{\text{demag}} = -\frac{1}{2} \int_V \mu_0 M_{\text{s}} (\mathbf{m} \cdot \mathbf{H}_{\text{demag}})\, dV. \tag{1.6}$$

For a uniformly magnetized spheroid the demagnetizing field in Eq. 1.6 simplifies to:

$$\mathbf{H}_{\text{demag}} = -N_{\text{shape}} M_{\text{s}} \mathbf{m} \tag{1.7}$$

Where N_{shape} is the demagnetization factor and describes the influence of the sample shape onto the demagnetization field. This shape-dependent demagnetizing field is what gives rise to shape anisotropy.

Shape Anisotropy

In thin-film soft ferromagnets, shape anisotropy governs the arrangement of easy and hard axes. As J. M. D. Coey describes in [8], the shape anisotropy energy is defined as the energy difference between magnetizing an object along its easy axis versus magnetizing it along its hard axis. Shape anisotropy can be characterized by the anisotropy constant K_{sh}, which describes the energy density required to rotate the magnetization from the easy axis to the hard axis.

$$K_{\text{sh}} = \frac{1}{2}\mu_0 M_{\text{s}}^2 (N_{\text{hard}} - N_{\text{easy}}) \tag{1.8}$$

N_{easy} is the demagnetizing factor associated the easy axis, and N_{hard} represents the demagnetizing factors associated with the hard axes. However as previously discussed a sample can have multiple hard and easy axis and they do not have to be orthogonal to each other. If the demagnetizing factors are calculated along three orthogonal axes, (x, y, z), the resulting demagnetization factors (N_x, N_y, N_z) sum up to 1:

$$N_x + N_y + N_z = 1 \tag{1.9}$$

For a sphere, $N = \frac{1}{3}$ in all directions, meaning that the demagnetizing field does not favor any specific axis. As a result, the shape anisotropy energy vanishes, since there is no preferred magnetization direction. Depending on the shape of the magnetic sample, demagnetizing factors can be determined using analytical expressions for highly symmetric shapes. Fig. 1.2 displays two example shapes: an elliptic cylinder and a rectangular prism. Both of them have well-defined analytical expressions for their demagnetizing factors. The exact mathematical formulations for these demagnetization factors are given in [13], [14] and elaborated upon in Appendix A.

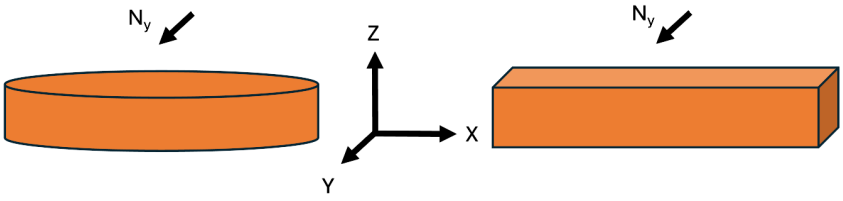


FIGURE 1.2: A rectangular prism and an elliptical cylinder are shown along with their spatial orientation. The figure also depicts the y-direction, along which the demagnetization factor N_{y} is determined.

The following will consider a rectangular prism defined by the three side lengths x_{len}, y_{len}, and z_{len}. For simplicity, the z_{len} will be kept constant, and the demagnetization factor will always be calculated along the y-direction, which is out of the plane of sight. The parameters x_{len} and y_{len} will be varied, with the main focus on the aspect ratio α_{xy}.

$$\alpha_{\text{xy}} = \frac{x_{\text{len}}}{y_{\text{len}}} \tag{1.10}$$

Fig. 1.3 enables an intuitive understanding of the demagnetization factors as well as the resulting easy and hard axes. Depicted is the analytically calculated demagnetization factor for varying sizes of a rectangular prism. In the following, three special cases are considered:

1. $x_{\text{len}} \approx y_{\text{len}}$ $\quad \alpha_{\text{xy}} \approx 1$
2. $x_{\text{len}} \ll y_{\text{len}}$ $\quad \alpha_{\text{xy}} \to 0$
3. $x_{\text{len}} \gg y_{\text{len}}$ $\quad \alpha_{\text{xy}} \to \infty$

1. On the diagonal from $(0,0)$ to $(1,1)$, x_{len} and y_{len} have approximately the same magnitude, and $N_y \approx 0.3$. Meaning that if $\alpha_{\text{xy}} \approx 1$ very little energy is required to change the magnetization from the x-direction to the y-direction. The sample is symmetric, and there is neither a distinct soft nor hard axis in the xy-plane.

2. If $x_{\text{len}} \ll y_{\text{len}}$, α_{xy} tends toward 0 and so does N_y. In this case, the demagnetization factor is calculated along the far longer direction of the rectangular prism. The demagnetization factor along this direction is minimal. Therefore, the longer side of the prism is also the easy axis of the sample. The magnetization favors an alignment along the longer sides of a shape. Also, Fig. 1.3 shows that the lower the ratio of x_{len} to y_{len}, the more N_y decreases. As a consequence of Eq. 1.9, this leads to a increase in N_x and N_z. which means that for rectangular prisms with a lower aspect ratio between the lengths in the x- and y-directions, the difference between the demagnetization factors in these directions increases. An increasing difference between the demagnetization factors enhances the shape anisotropy, making it energetically more expensive to rotate the magnetization out of the easy axis.

3. If the same shape is rotated and $x_{\text{len}} \gg y_{\text{len}}$, N_y tends toward 1 and the aspect ratio tends towards infinity. Showcasing the symmetric behavior of the demagnetization factor.

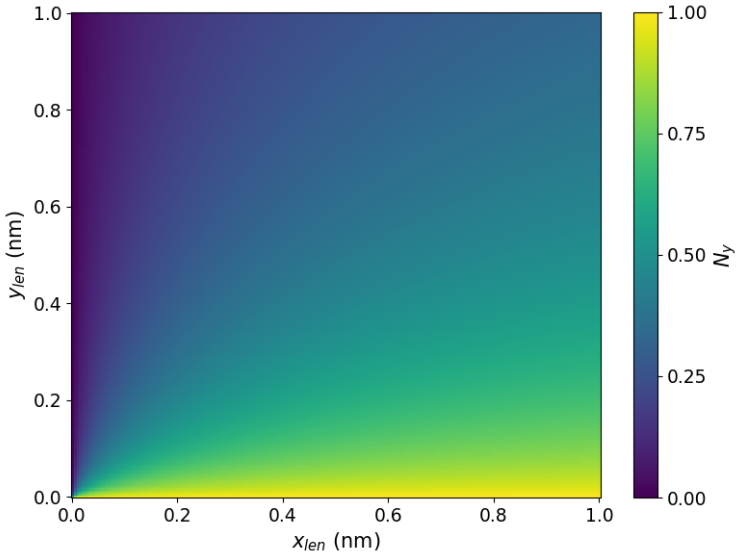
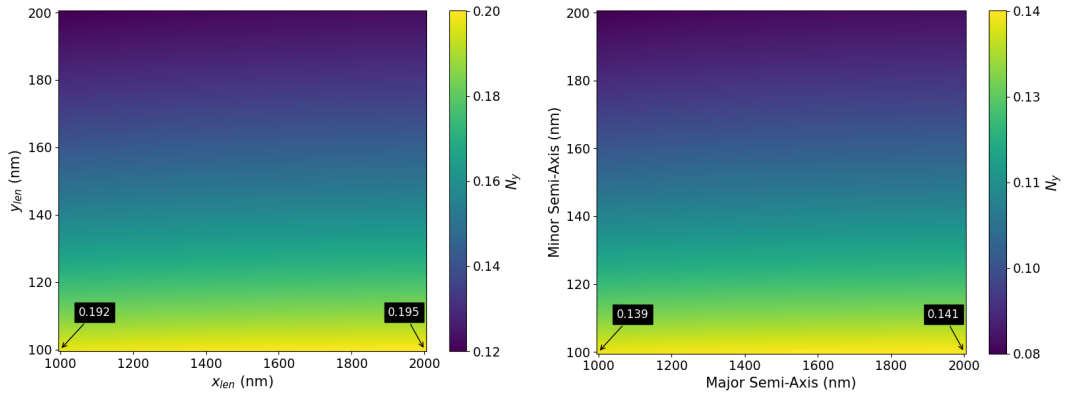


FIGURE 1.3: Demagnetization factor N_y along the y-axis, for a rectangular prism. The length of the sample along the z-axis is 1 nm, while x_{len}, y_{len} are varied. The analytical expressions for these demagnetization factors are provided in Appendix A.

Fig. 1.4 displays the N_{y} for an elliptical cylinder as well as a rectangular prism. $z_{\text{len}} = 20\,\text{nm}$ is again kept constant, and the x_{len}, y_{len} as well as the major and minor axes are varied in the interval $[100, 200] \times [1000, 2000\,]$(nm). The depicted shapes are chosen to be representative of those used in tunneling magnetoresistance sensors, as elaborated in Sec. 2. They also showcase the characteristics elucidated from Fig. 1.3, where the largest ratio of x_{len}, y_{len} leads to the highest demagnetization factor. However, since y_{len} is an entire order of magnitude higher than x_{len}, the difference is small, resulting in a difficult numerical optimization problem.



(A) Rectangular Prism (B) Elliptic Cylinder

FIGURE 1.4: Demagnetization factor along the y-axis, N_y, for a rectangular prism in (A) and an elliptical cylinder in (B). The length of the sample along the z-axis is 20 nm, while x- and y-lengths, as well as the major and minor axes, are varied in the interval $[100, 200] \times [1000, 2000]\,(nm)$. The exact calculations of these demagnetization factors are described in A.

The hysteresis curve is significantly influenced by the demagnetization factor, particularly in soft magnets [12]. When the external field is reversed, the magnetization rotates to follow the field. A higher aspect ratio leads to a higher energy requirement to rotate the magnetization into its hard axis. Therefore, the rotation is spread over a broader range of the external field. Consequently, the transition of the hysteresis curve has a smaller slope.
To illustrate the influence of aspect ratios on the hysteresis curve, Fig. 1.5 is designed. Depicted there are the theoretical hysteresis curves for rectangular prisms with different aspect ratios. A theoretical hysteresis curve can be approximated with three assumptions:

1. Homogeneous magnetization throughout the entire sample.
2. Homogeneous demagnetization fields.
3. The magnetization rotates only in the xy-plane and has no z-component.

With these three assumptions, Eq. 1.2 of the free energy can be reformulated to express the equilibrium state of m_y as a function of the external field. A detailed description of this reformulation is given in Appendix B.

TABLE 1.1: Three rectangular prisms with different lengths in the x and y directions, as well as their aspect ratios α_{xy}, are considered. Each prism has the length of 100 nm along the z- direction. The analytical expressions for the demagnetization factors N_x and N_y are provided in Appendix A, and the approximation of the hysteresis curve in Appendix B. The color code refers to the hysteresis curves in Fig. 1.5.

x_{len}, y_{len} (nm)	α_{xy}	N_x, N_y	Slope of Hysteresis Loop	Color Code Fig. 1.5
1.001, 0.999	1.002	0.098, 0.097	1499	blue
2, 1	2	0.108, 0.052	18.1	orange
10, 1	10	0.118, 0.011	9.3	green

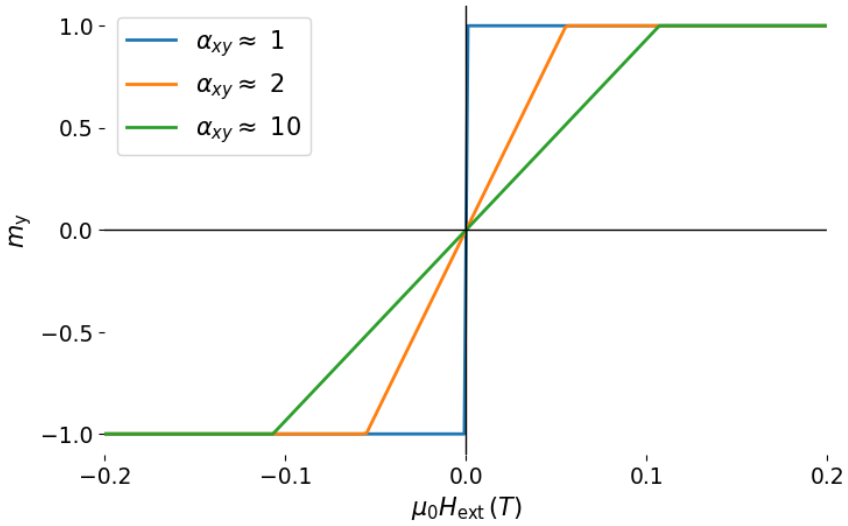


FIGURE 1.5: Approximate hysteresis curves for three different rectangular prisms. Their specifications can be found in Tab. 1.1. All of them have a length of 100 nm in the z-direction. The analytical approximation of the hysteresis loops is given in Appendix B.

As depicted in Fig. 1.5, a increase in the aspect ratio of rectangular prisms enhances shape anisotropy effects, leading to a lower slope in the hysteresis loop. Section 5.3 presents a numerical investigation of these findings and discusses their generalization to additional shapes.

1.3 Numerical Micromagnetics

The previous section discusses the free energy of continuous samples. Analytical solutions for the free energy can be found for primitive shapes [15]. For complex shapes, numerical approximations have to be considered. Therefore, approaches like the finite element method (FEM) are used to approximate these continuous problems [16]. The approach is to break down complex objects into small subdivisions. These finite elements are usually tetrahedrons or hexahedrals and will be referred to here simply as mesh cells. However, to solve the free energy for these finite elements, the formulation of the total free energy developed in Sec. 1.2 also has to be discretized. Therefore, the magnetization unit vector $\mathbf{m}$ is linearly interpolated in each finite element n of a mesh and described as $\mathbf{x} \in \mathbb{R}^{3n}$. Here, $\mathbf{x}$ contains the three spatial coordinates of the magnetization for each unit cell.

$$\mathbf{x} = \begin{bmatrix} m_{1,x} & m_{1,y} & m_{1,z} & m_{2,x} & m_{2,y} & m_{2,z} & \dots & m_{n,x} & m_{n,y} & m_{n,z} \end{bmatrix}^{\mathrm{T}} \tag{1.11}$$

$$|\mathbf{x}_i| = \sqrt{m_{i,x}^2 + m_{i,y}^2 + m_{i,z}^2} = 1, \quad \text{for } i = 1, \dots, n \tag{1.12}$$

$\mathbf{m}$ is subject to the micromagnetic constraint given in Eq. 1.12, meaning that its length is fixed at unity, and it only rotates its orientation. In comparison to the magnetization, the external field $\mathbf{h}_{\text{ext}}$ and the demagnetization field $\mathbf{h}_{\text{d}}$ do not have a constant magnitude. Nevertheless, they can be vectorized at the same mesh points as the magnetization and represented as vectors in $\mathbb{R}^{3n}$. This notation enables the numerical approximation of the free energy in Eq. 1.13. The matrix $\mathbf{C}$ encodes grid information and intrinsic properties of the sample to account for the exchange interaction as

well as the anisotropy contributions in $\frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{C}\mathbf{x}$. The entries in the matrix $\mathbf{C}$ account for the local energy contributions across the sample. $\mathbf{M}$ represents local variations in the spontaneous magnetization are represented as a vector field, capturing the interaction between the magnetic field and magnetization. $\frac{1}{2}\mathbf{h}_{\mathrm{d}}^{\mathrm{T}}\mathbf{M}\mathbf{x}$ describes the influence of the demagnetization field, resulting in the demagnetization energy, and $\mathbf{h}_{\mathrm{ext}}^{\mathrm{T}}\mathbf{M}\mathbf{x}$ describes the influence of an external field, resulting from the Zeeman energy. This notation closely follows [17], where more precise formulations of $\mathbf{M}$ and $\mathbf{C}$ can be found.

$$\mathcal{E}_{\mathrm{tot}}(\mathbf{x}) = \underbrace{\frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{C}\mathbf{x}}_{\text{(I)}} - \underbrace{\frac{1}{2}\mathbf{h}_{\mathrm{d}}^{\mathrm{T}}\mathbf{M}\mathbf{x}}_{\text{(II)}} - \underbrace{\mathbf{h}_{\mathrm{ext}}^{\mathrm{T}}\mathbf{M}\mathbf{x}}_{\text{(III)}} \tag{1.13}$$

The exchange energy as described in Eq. 1.4, favors parallel spin alignment. Similarly, the anisotropy energy $\mathcal{E}_{\mathrm{a}}$ penalizes deviations from the easy axis. In this case, (I) attains its minimum when the magnetization aligns perfectly with the easy axis, ensuring no misalignment between neighboring spins. The Zeeman energy in (III), favors an alignment with the external field. If the external field is not aligned with the easy axis, the Zeeman and anisotropy energy terms compete. The contribution in (II) arises from the demagnetization field H_{demag}. Because H_{demag} is a function of the magnetization distribution, changes in magnetization continuously adjust H_{demag}, creating an iterative, self-consistent feedback loop in the minimization process. The term is minimized when the net magnetization decreases, for example, through non-uniform alignment, which may counteract the uniform alignment preferred by term (I). As described in Sec. 1.2.4, H_{demag} forms in the opposite direction of the external field, implying that (II) and (III) also counteract each other. Each configuration of the magnetic moments results in a specific energy balance. Naturally, the configuration with the lowest total energy is favored and defines the equilibrium state. The competing energy terms described above are leading to an non-trivial equilibrium configuration.

1.4 Numerical Energy Minimization

As discussed in Sec. 1.3, the objective is to determine an $\mathbf{x}$ that minimizes the energy balance. This is a numerical optimization task, and several approaches exist, focusing on improving convergence speed, stability, and computational efficiency. Historically, the limited-memory Broyden-Fletcher–Goldfarb-Shanno method (LBFGS) [18] and the Conjugate Gradient (CG) method [19] have been most commonly used. Recent advancements in physics-informed machine learning methods have yielded significant improvements and promising outcomes [20].
J. Fischbacher et al. tested several variations of the CG and LBFGS solvers in [17] on soft magnetic thin-film elements similar to those investigated in this study. They found that both methods can accurately compute the equilibrium states and exhibit similar convergence behavior and efficiency. This study focuses on the CG method due to its low memory requirements and ability to utilize GPUs for efficient calculation [21].

Conjugate Gradient Method

To enable an intuitive approach to the Conjugate Gradient Method, a basic version of the algorithm will be discussed first. Upon these foundations, modifications will be introduced to adapt the method for micromagnetic simulations and improve computational robustness and efficiency. This section is based on [22].

In the most simplified version, the CG method is aimed to solve a system of linear equations like $\mathbf{C}\mathbf{x} = -\mathbf{b}$. Equivalently, it minimizes the quadratic function $Q(\mathbf{x}) = \frac{1}{2}\mathbf{x}^{\mathrm{T}}\mathbf{C}\mathbf{x} + \mathbf{b}^{\mathrm{T}}\mathbf{x} + \mathbf{c}$ with respect to the vector $\mathbf{x}$. Here, $\mathbf{C}$ is the positive symmetric problem matrix, $\mathbf{b}$ is the solution vector, and $\mathbf{c}$ is the bias term.

Algorithm 1 Conjugate Gradient Method

1: **Input:** Matrix $\mathbf{C}$, vector $\mathbf{b}$, tolerance ϵ
2: Initialize $\mathbf{x}_0$, $\mathbf{g}_0 = \mathbf{C}\mathbf{x}_0 + \mathbf{b}$
3: Set $\mathbf{d}_0 = -\mathbf{g}_0$
4: **for** $j = 0, 1, 2, \ldots$ **do**
5: $\quad \alpha_j = \frac{\mathbf{g}_j^{\mathrm{T}}\mathbf{g}_j}{\mathbf{d}_j^{\mathrm{T}}\mathbf{C}\mathbf{d}_j}$
6: $\quad \mathbf{x}_{j+1} = \mathbf{x}_j + \alpha_j\mathbf{d}_j$
7: $\quad \mathbf{g}_{j+1} = \mathbf{g}_j + \alpha_j\mathbf{C}\mathbf{d}_j$
8: $\quad$ **if** $\|\mathbf{g}_{j+1}\| < \epsilon$ **then exit**
9: $\quad$ **end if**
10: $\quad \beta_j = \frac{\mathbf{g}_{j+1}^{\mathrm{T}}\mathbf{g}_{j+1}}{\mathbf{g}_j^{\mathrm{T}}\mathbf{g}_j}$
11: $\quad \mathbf{d}_{j+1} = -\mathbf{g}_{j+1} + \beta_j\mathbf{d}_j$
12: **end for**

Algorithm 1 presents the fundamental formulation of the CG method. It begins in line 2 with an initial guess $\mathbf{x_0}$ and the gradient $\mathbf{g_0}$ at the initial guess. Since the gradient always points in the direction of the steepest ascend, the first search direction $\mathbf{d_0}$ is set opposite to the gradient. The iterative loop of Algorithm 1 repeats until the stopping criterion in line 8 is met. In this case, convergence is considered achieved if the gradient norm falls below a given tolerance ϵ. At the beginning of each iteration (line 5), the optimal step length α_j is computed analytically using a line search. Based on the computed optimal step length, the variables $\mathbf{x}$ and $\mathbf{g}$ are updated in the lines 6 and 7. If convergence is not reached, the conjugate factor β_j is computed in line 10. This factor constitutes the core of the CG method, ensuring that each search direction is $\mathbf{C}$-conjugate (Eq. 1.14) to all previous directions. Under linear conditions, this structure leads to convergence within at most n iterations, provided that $\mathbf{C}$ is full-rank and n denotes its rank. This orthogonality arises from how the method constructs its directions within appropriately formed Krylov subspaces. For a more detailed explanation, see [23].

$$\mathbf{d}_i^{\mathrm{T}}\mathbf{C}\mathbf{d}_j = 0, \quad \text{for all } i \neq j \tag{1.14}$$

Specially the micromagnetic constraint in Eq. 1.12 introduces strong nonlinearity to the total free energy. Therefore, Alg. 1 has to be modified. These modifications are applied in Alg. 2. Initially, in line 3, the magnetic configuration $\mathbf{x}_0$ must be chosen with a unit norm to obey the micromagnetic constraint in Eq. 1.12. Two mechanisms are used to ensure that the micromagnetic constraint is enforced in every iteration

of the optimization. The initial gradient in line 3, as well as each update of the gradient in line 8, is projected onto the orthogonal projection$(\Pi_{\mathbf{x}^\perp}\mathbf{v})^{(i)}$ of the magnetic configuration.

$$(\Pi_{\mathbf{x}^\perp}\mathbf{v})^{(i)} = \mathbf{v}^{(i)} - \left((\mathbf{v}^{(i)})^{\mathrm{T}}\mathbf{x}^{(i)}\right)\mathbf{x}^{(i)} \tag{1.15}$$

This projection ensures that magnetic moments are only rotated and do not change their magnitude. The second measure to obey the micromagnetic constraint is that whenever the micromagnetic configuration is updated, it is also normalized with $(\mathcal{N}\mathbf{v})^{(i)} = \mathbf{v}^{(i)}/\left|\mathbf{v}^{(i)}\right|$. Therefore, $\mathcal{N}$ is applied to the updates of $\mathbf{x}$ in lines 6 and 7.

Algorithm 2 Preconditioned Nonlinear Conjugate Gradient for Micromagnetics

1: **Task:** Minimize $\mathcal{E}_{\text{tot}}(\mathbf{x})$ subject to $|\mathbf{x}^{(i)}| = 1$.
2: **Initialize:**
3: Start with $\mathbf{x}_0$ such that $|\mathbf{x}_0^{(i)}| = 1$ and compute $\mathbf{g}_0 = \Pi_{\mathbf{x}_0^\perp}(\nabla F(\mathbf{x}_0))$
4: Solve $\mathbf{P}_0\mathbf{y}_0 = \mathbf{g}_0$, set $\mathbf{d}_0 = -\mathbf{y}_0$
5: **for** $j = 0, 1, 2, \ldots$ **do**
6: Compute α_j by minimizing $F(\mathcal{N}(\mathbf{x}_j + \alpha_j\mathbf{d}_j))$
7: Update $\mathbf{x}_{j+1} = \mathcal{N}(\mathbf{x}_j + \alpha_j\mathbf{d}_j)$
8: Compute $\mathbf{g}_{j+1} = \Pi_{\mathbf{x}_{j+1}^\perp}(\nabla F(\mathbf{x}_{j+1}))$
9: Solve $\mathbf{P}_{j+1}\mathbf{y}_{j+1} = \mathbf{g}_{j+1}$
10: Update $\mathbf{d}_{j+1} = -\mathbf{y}_{j+1} + \beta_j^*\mathbf{d}_j$
11: **end for**

Alg. 2 implements a preconditioned version of the CG Method. The fundamental idea is to transform the problem using a symmetric positive definite (SPD) preconditioning matrix $\mathbf{P}$ to improve the conditioning number κ of the system.

$$\mathbf{P}^{-1}\mathbf{C}\mathbf{x} = -\mathbf{P}^{-1}\mathbf{b} \tag{1.16}$$

$$\kappa(\mathbf{P}^{-1}\mathbf{C}) \ll \kappa(\mathbf{C}). \tag{1.17}$$

To utilize these numerical advantages, the initial search direction, as well as each update of the search direction, is defined as the negative preconditioned residual $\mathbf{y}$.

$$\mathbf{y} = \mathbf{P}^{-1}(\mathbf{C}\mathbf{x} + \mathbf{b}). \tag{1.18}$$

To determine these preconditioned residuals, the linear system $\mathbf{P}\mathbf{y} = \mathbf{g}$ has to be solved in each iteration of the algorithm (line 10) as well as in the initial setup (line 3). Solving linear systems, however, is comparatively computationally inexpensive. L. Exl et al. [22] recommends the usage of an approximated Hessian as a preconditioner, which better captures the curvature of the energy landscape. With this approach, they achieve a speedup factor of up to 3 for simulations of soft magnets. Another difference in considering a nonlinear problem is that the optimal step length cannot be determined as quickly as in the linear case. Instead of a simple matrix-vector multiplication in line 5 of Alg. 1, the nonlinear case requires minimization of $F(\mathcal{N}(\mathbf{x}_j + \alpha_j\mathbf{d}_j))$. In practice, the exact solution of this minimization is too computationally expensive. In [17] Fischbacher et al, examines different approximations of the optimal step length and demonstrates that a limited approximation of the step length is computationally feasible and results in the fastest convergence. They also

investigates different versions of the conjugate gradient factor β, showcasing that a modification of the Hestenes-Stiefel [19] method, resulting in a conjugate factor β_i^*, is most effective for soft magnetic thin films [17].

$$\beta_i^* = \mathbf{d}_{i+1} = \begin{cases} \beta_i^{\mathrm{HS}} & \text{if } \|\mathbf{g}_{i+1}\|^2 > |\mathbf{g}_{i+1}^{\mathrm{T}}\mathbf{g}_i| \\ 0 & \text{otherwise} \end{cases} \tag{1.19}$$

$$\beta_i^{\mathrm{HS}} = \frac{(\mathbf{g}_{i+1} - \mathbf{g}_i)^{\mathrm{T}}\mathbf{g}_{i+1}}{(\mathbf{g}_{i+1} - \mathbf{g}_i)^{\mathrm{T}}\mathbf{d}_i} \tag{1.20}$$

Chapter 2

Tunneling Magnetoresistance Sensors

Tunneling magnetoresistance (TMR) sensors are designed to measure an external magnetic field via electric resistance. The need for such magnetic field sensors is rapidly increasing, especially as the Internet of Things (IoT) aims to connect more and more autonomous sensors to the internet and enable smart applications with their data. Their applications extend across various domains, from current sensing and temperature monitoring to advanced biomedical detection [24].

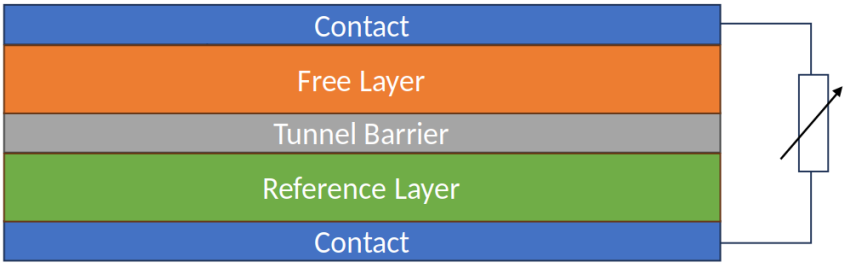


FIGURE 2.1: Schematic structure of a Tunnel Magnetoresistance (TMR) sensor. The device consists of a Free Layer, Tunnel Barrier, and Reference Layer, sandwiched between Contact electrodes. The magnetization of the Free Layer is variable, whereas the Reference Layer maintains a fixed magnetization, enabling resistance changes based on their relative alignment.

They rely on the concept of two ferromagnetic layers which are separated by a non-conducting tunneling barrier. The fundamental structure is illustrated in Fig. 2.1. The first ferromagnetic layer, called the Reference Layer, has a pinned magnetization, while the second ferromagnetic layer, called the Free Layer, has a magnetization that freely aligns with an external field. In 1975, M. Jullière [25] was the first to experimentally demonstrate that the conductivity between these two layers depends on the angle between the magnetizations of the Free and Reference Layer. Which enables the measurement of the orientation of the external field relative to the orientation of the Reference Layer. The second major milestone in the development of TMR sensors was achieved in 1995, when T. Miyazaki [26] developed TMR sensors operating at room temperature, thus opening up a broad range of scientific and industrial applications.

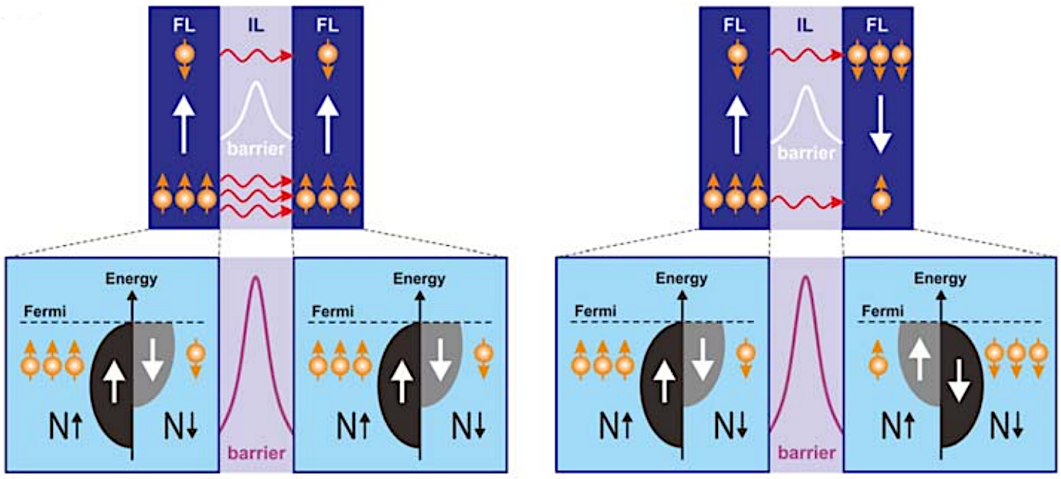
Unless stated otherwise, this Chapter is based on the review article by N. A. Wibowo et al. [4], which discusses the fundamentals and potential of tunneling magnetoresistance.

2.1 Density of States in an External Magnetic Field

TMR sensors are based on the principle that:

> The tunneling probability between two ferromagnets, separated by an insulating barrier, depends on the orientation of their magnetization.

In classical physics, electrons would not be able to pass through the insulating barrier at all. The Schrödinger equation, however, describes particles with a wave-like nature. Therefore, electrons have a nonzero probability of being found on the other side of the barrier. This phenomenon is called quantum tunneling, and its probability depends on both the insulating layer and the state of the two ferromagnets. The insulating layer influences the tunneling probability through its thickness and the magnitude of the resulting tunneling barrier. These barriers are sketched in Fig. 2.2.



(A) High Conductivity in Parallel State (B) Low Conductivity in Antiparallel State

FIGURE 2.2: Electron tunneling in a magnetic tunnel junction (MTJ) for two magnetization configurations. (A) Parallel configuration: Majority-spin electrons tunnel through the insulating barrier into an identical spin density of states, leading to high conductivity. (B) Anti-parallel configuration: Electrons tunnel into a non-identical spin density of states, resulting in lower conductivity. (FL: ferromagnetic layer, IL: insulating layer). Adapted from [4], licensed under CC-BY 4.0 https://creativecommons.org/licenses/by/4.0/.

The ferromagnetic layers influence the tunneling probability through their magnetization. In a ferromagnet, the exchange interaction splits the electronic band structure into spin-up and spin-down subbands. An external field aligns the net magnetization, resulting in a spin polarization that shows a 'majority spin' subband (parallel to the field) and a 'minority spin' subband (antiparallel to the field). Fig. 2.2 schematically illustrates how majority (black) and minority (gray) subbands differ in occupancy, depending on the magnetization direction. Under the spin-conserving

tunneling assumption, electrons are assumed not to change their spin while traveling through the tunneling barrier. As a result, electrons are more likely to tunnel through the barrier if the density of states is similar on both sides of the barrier. This is also sketched in Fig. 2.2, where parallel states of the two ferromagnetic layers lead to more electrons traveling through the tunneling barrier compared to the anti-parallel state.

2.2 Challenges and Optimization

In practice, the manufacturing of TMR sensors presents many challenges. For example, high sensitivity, meaning an enhanced difference in resistance between the parallel and antiparallel magnetization states, is important to achieve a proper signal-to-noise ratio. Another goal for TMR sensors is to cover a broad field range and provide a linear and uninterrupted response over an increased range. Especially in IoT applications, low power consumption and enhanced temperature stability are also key considerations.
In [27] L. A. Francis and K. Poletkin describes the voltage change ΔV of a TMR sensor as follows:

$$\Delta V = \frac{1}{2}\mathrm{TMR}\, i\, \cos(\theta_p - \theta_f)\, \frac{R\,A}{Wh}. \tag{2.1}$$

Where TMR is the maximum tunnel magnetoresistance level, i the bias current, $R \times A$ the resistance per area parameter, and Wh the width and thickness of the sensor element. θ_p and θ_f are the magnetization angles of the pinned and free layers, respectively. While θ_p remains constant, the magnetization of the free layer adjusts to the external field. The magnetization of a sample for a given external field is described by the hysteresis curve. Therefore, the hysteresis curve of the free layer fundamentally influences the voltage change in a TMR sensor.
The hysteresis curve of a sample depends on several different criteria, like the shape anisotropy or the material properties. An optimization of this hysteresis curve must always be performed based on the specific application use case. For example, it is not possible to generalize which of the three approximated hysteresis curves in Fig. 1.5 is the most advantageous for every application. In general, a smaller slope of the hysteresis curve of the free layer leads to a broader field range for the sensor. However, if the slope is too low, the signal-to-noise ratio could hinder the desired measurements. As a result of the different use cases of TMR sensors, simulation frameworks are needed to provide easy access to optimize magnetic devices for a variety of different problems. Research projects, such as the Magnetic Multiscale Modeling Suite [1], aim to provide this simulation framework as open-source, user-friendly software for optimizing a broad range of applications.

Chapter 3

Bayesian Optimization

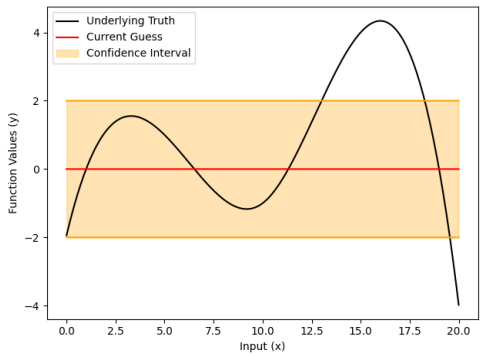
Within this work, various sensor shapes and sizes will be explored. Each of them requires micromagnetic simulation to determine their characteristics. These simulations are typically performed on objects consisting of over 20 million finite mesh elements. Even without considering the meshing, energy minimization for such large systems often requires several days [22]. Thus, an optimization algorithm with high sampling efficiency for newly acquired function evaluations is essential.
Determining the magnetization of a soft magnet under a specific external field, as outlined in Section 1, is a highly complex process. Hence, an optimizer is employed that treats the entire simulation process as a black box. The optimizer has access solely to the sensor's shape parameters and the corresponding objective values previously obtained as labels for these parameters.
A Bayesian optimizer will be employed in this work to identify optimal sensor geometries. This method is particularly advantageous due to its high sample efficiency in complex, high-dimensional search spaces and its suitability for black-box optimization [28].

The following section is primarily based on the Proceedings of the 2021 Winter Simulation Conference and follows the main ideas presented in the publication "A Gentle Introduction to Bayesian Optimization" by Candelieri and Antonio (2021) [29], as well as the foundational book "Bayesian Optimization" by Garnett (2023) [30].

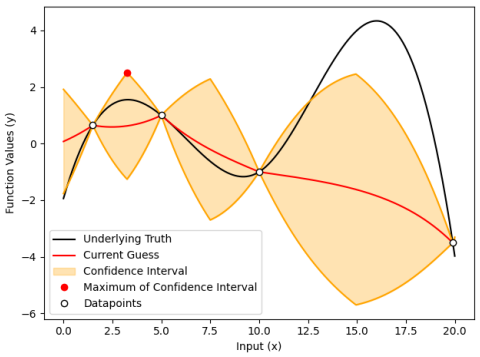
3.1 Introduction to Bayesian Optimization

In use cases where both the optimal solution and the cost of optimization must be considered, it is advantageous to optimize a simplified model function instead of the expensive objective function. This is where the Bayesian Optimizer (BO) becomes essential. Rather than directly optimizing the costly objective function, BO utilizes Gaussian Processes to construct a probabilistic surrogate model that approximates the objective function and is computationally inexpensive to maximize [29]. Utilizing this model, acquisition functions are employed to determine the most suitable next sample for function evaluation. Each of these data points further enhances the understanding of the objective function and leads to an update of the statistical model [31].
To enable an intuitive approach, the following will outline the fundamental ideas of Bayesian Optimization. The aim is to introduce the core principles of optimization in an accessible manner, while a more detailed and mathematically precise discussion is reserved for Sec. 3.2.

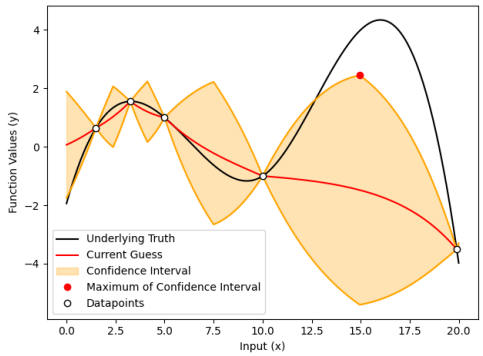
In the following example, the aim of the optimization is to find the maximum of the objective function, or the so-called "underlying truth," presented in Fig. 3.1A in black. This function is initially unknown and will only be revealed in the process of the optimization through single function evaluations. The x-axis represents the input variable, while the y-axis denotes the function values of the objective function.
Before any data points are considered, there are already assumptions that can be made for many optimization problems. Even if the underlying truth is unknown, the general problem usually is not. For example, while the exact demagnetizing fields of a given ferromagnet may be unknown, prior research allows for an estimation of their order of magnitude. Therefore, for the given example, an initial guess is set to be 0, and the interval in which data points are likely to be observed is guessed to be around $[-2, 2]$. This confidence interval does not need to be correct or contain all the values of the underlying truth but rather act as a starting point for the iterative optimization. This initial guess forms the foundation for the prior distribution described in Sec 3.2.
After the initial guess, data points can be considered. In Fig. 3.1B, four random placed data points have been added. These data points influence the current guess and adjust the confidence interval accordingly. The current guess now aligns with the data points, resulting in variations from the initial value of 0. Additionally, the confidence interval is 0 at the data points since the function is known, leaving no uncertainty. If there are error margins associated with the data points, the confidence interval at the data points could be larger. The confidence interval also increases with distance from the data points since the variation of the objective function could increase. This incorporation of data points will be introduced in Sec. 3.2 as the likelihood function.
After the adjustments resulting from the data points, an "educated guess" for the optimal position of the next data point can be made. This is where the acquisition functions come into play. The acquisition function determines which sample point should be selected next. They will be based on the confidence interval and current guess to suggest the position of the next data point. These acquisition functions will be further discussed in Sec. 3.3. For simplicity, the example in Fig. 3.1 uses an acquisition function that selects the next data point where the upper bound of the predictive confidence interval is maximal. This maximum is marked as a red point in Fig. 3.1B. During the next iteration, the optimizer evaluates the suggested data point, leading to an updated prediction. The resulting best guess, as well as the confidence interval after considering 5 data points, is shown in Fig. 3.1C. However, with the knowledge of the underlying truth in mind, it is evident that the fifth data point at $x \approx 3$ is part of a local maximum and not the global maximum at $x \approx 17$. After this local maximum is exploited, the new maximum of the confidence interval is in the region of the global maximum. The uncertainty of this unexplored area outweighs the already known data points at the global maximum. This balance between exploitation the current best and exploration uncertain areas demonstrates the ability of Bayesian Optimization to escape local maxima and find global ones. As shown in Fig. 3.1D, the next three data points will be set close to the global maximum at $x \approx 17$. Fig. 3.1D also presents an example of a stopping criterion: after 8 data points, the maximum of the confidence interval does not exceed the data points, which indicates the end of the optimization in this example.



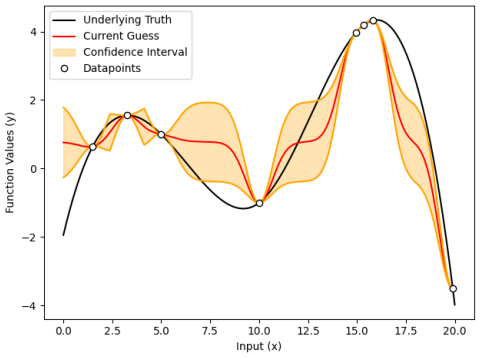
(A) Bayesian Optimization with Initial State



(B) Bayesian Optimization with 4 Sample Points



(C) Bayesian Optimization with 5 Sample Points



(D) Bayesian optimization with 8 sample points

FIGURE 3.1: Examples of four states of Gaussian optimization. In black, the objective function or so-called "underlying truth." In red, the current guess of the objective function. In orange, the confidence interval where values of the objective function are assumed. The red dot in (B) and (C) represents the maximum of the confidence interval. (A) represents the initial state without considering any data points. (B) considers 4 data points, the updated current guess, and the updated confidence interval. The maximum of the confidence interval, shown as the red point. (C) considers 5 data points, and the maximum of the confidence interval is related to the global maximum. (D) considers 8 data points and is the final state of the optimization.

3.2 Gaussian Processes

Initially, every optimization needs an objective function. In this context, this will be f defined on the infinite domain $\mathcal{D}$.

$$f : \mathcal{D} \to \mathbb{R} \tag{3.1}$$

The aim of optimization is, for example, to identify the global maximum x^* of the objective function.

$$x^* \in \arg\max f(x) \tag{3.2}$$

Thus, by evaluating data points, the unknown function f is progressively revealed with each additional data point x. To combine the knowledge from different data points, Gaussian Processes are used. However, since the aim is not to gain the most possible knowledge about f but to actually maximize it, each new data point has

to be chosen wisely. What this means mathematically will be explained in Sec 3.3, where so-called acquisition functions make the most use of the knowledge gained from the Gaussian Processes.

A Gaussian Process is used to model complex objective functions through multivariate normal distributions. Therefore, f is defined as nonparametric and can be conceptualized as an infinite ensemble of random variables. Each random variable corresponds to the function's value at a specific point in the domain, and the mutual dependencies between these variables determine the statistical properties and shape of the function [30]. This implies that for any set of points $\mathbf{x} = \{x_1, x_2, \ldots, x_n\} \in \mathcal{D}^n$, a Gaussian Process defines joint Gaussian distributions over the objectiv function values
$\boldsymbol{\phi} = \{\phi_1, \phi_2, \ldots, \phi_n)\} \in \mathbb{R}^n$. These joint Gaussian distributions provide a consistent probabilistic model over the entire domain $\mathcal{D}$, even if they are defined based on a finite subset of points $\{x_1, x_2, \ldots, x_n\}$ [30].

3.2.1 Initial beliefs as a Prior Distribution

Before any data points are collected, initial beliefs about f are modeled as a Gaussian Process called the prior distribution $p(\boldsymbol{\phi}|\mathbf{x})$. The so-called "prior" is the conditional probability of ϕ at a location $\mathbf{x}$ and depends on three things. First, the objective function f. Secondly, the mean function $\mu : \mathcal{D} \to \mathbb{R}$, which represents the expected value of the distribution of f at each point considered in the input space.

$$\mu(x_i) = \mathbb{E}[f(x_i)] \tag{3.3}$$

Thirdly, the covariance function $K : \mathcal{D} \times \mathcal{D} \to \mathbb{R}$, which can be described as a kernel, representing correlations between function values at different points x_i, x_j. Therefore, K encodes the expected behavior and properties of f as structured deviations from the mean values. These deviations enable the concept of the confidence interval, which describes the range that captures the predicted value of $f(x)$ with 95% probability. For a Gaussian Process, this can be expressed as:

$$\text{Confidence Interval} = \mu(x) \pm 0.95 \cdot \sigma(x),$$

where $\sigma(x)$ is the standard deviation derived from the covariance function K. The covariance function, often referred to as the kernel function, varies based on the use case and requires hyperparameter tuning for optimal function approximation.

$$K(x_i, x_j) = \text{cov}[f(x_i), f(x_j) \mid x_i, x_j] \tag{3.4}$$

To represent the expected value of whole sets, the vector notation for $\mathbf{x}$ is introduced:

$$\boldsymbol{\mu} = \mathbb{E}[f(\mathbf{x})] = \mu(\mathbf{x}), \tag{3.5}$$
$$\Sigma = \text{cov}[f(\mathbf{x}) \mid \mathbf{x}]. \tag{3.6}$$

With this vector notation in place, the prior belief before observing data can be summarized as:

$$p(\boldsymbol{\phi}|\mathbf{x}) = \mathcal{N}(\boldsymbol{\phi}; \boldsymbol{\mu}, \Sigma). \tag{3.7}$$

Where $\mathcal{N}$ is the multivariate Gaussian distribution:

$$\mathcal{N}(\boldsymbol{\phi}; \boldsymbol{\mu}, \Sigma) = \frac{1}{\sqrt{(2\pi)^d |\Sigma|}} \exp\left(-\frac{1}{2}(\boldsymbol{\phi} - \boldsymbol{\mu})^{\mathrm{T}} \Sigma^{-1} (\boldsymbol{\phi} - \boldsymbol{\mu})\right). \tag{3.8}$$

3.2.2 Data Points and the Likelihood Function

To develop a model that not only considers prior beliefs about a problem but also incorporates measured data, a set of observations of f at any point $x_i \in \mathcal{D}$ is considered. These observations result in a measured value $y_i \in \mathbb{R}$. These measured values can differ from the objective function for several reasons, such as measurement inaccuracies or imperfect simulations. Therefore, the value of the objective function is described as $\phi_i = f(x_i)$ and represents the measurement disturbed by some Gaussian distributed error ϵ:

$$f(x_i) = \phi_i = y_i + \epsilon, \tag{3.9}$$

$$\epsilon \sim \mathcal{N}_d(0, \sigma^2). \tag{3.10}$$

Where $\mathcal{N}_d(0, \sigma^2)$ represents a univariate Gaussian distribution and σ denotes the standard deviation of the noise. Given the objective function ϕ, the probability of observing the value y_i at the point x_i can be described using a Gaussian model.

$$p(y_i \mid x_i, \phi_i) = \mathcal{N}(y_i; \phi_i, \sigma^2). \tag{3.11}$$

Eq. 3.11 can again be vectorized to:

$$p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\phi}) = \prod_i p(y_i \mid x_i, \phi_i), \tag{3.12}$$

$$p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\phi}, \sigma_n) = \mathcal{N}(\mathbf{y}; \boldsymbol{\phi}, \sigma_n^2). \tag{3.13}$$

$p(\mathbf{y} \mid \mathbf{x}, \phi)$ is known as the likelihood function and represents the probability of observing a given data point.

3.2.3 Posterior Distribution

The initial beliefs contained in the prior distribution $p(\phi \mid \mathbf{x})$, combined with the observed data points in the likelihood function $p(\mathbf{y} \mid \mathbf{x}, \phi)$, build the posterior distribution. Vividly, it expresses the prior weighted by the likelihood function. The posterior distribution is described through Bayes' theorem:

$$p(\boldsymbol{\phi} \mid \mathbf{x}, \mathbf{y}) = \frac{p(\boldsymbol{\phi} \mid \mathbf{x}) p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\phi})}{p(\mathbf{y} \mid \mathbf{x})}. \tag{3.14}$$

To ensure that the posterior distribution is scaled to 1, it is normalized by the marginal likelihood $p(y \mid x)$ presented in Eq. 3.15.

$$p(\mathbf{y} \mid \mathbf{x}) = \int p(\mathbf{y} \mid \mathbf{x}, \boldsymbol{\phi})\, p(\boldsymbol{\phi} \mid \mathbf{x})\, d\phi \tag{3.15}$$

Therefore, the posterior distribution provides a foundation for further optimization. It quantifies the uncertainty resulting from the prior distribution as well as the information derived from the known observation points in the likelihood function. The posterior distribution enables the acquisition function to select advantageous next

sampling points to maximize the objective function. For notational convenience, the posterior distribution $p(\boldsymbol{\phi} \mid \mathbf{x}, \mathbf{y})$ will henceforth be denoted as p_{post}.

3.3 Acquisition functions

As described above, all the available information coming from prior knowledge or evaluated data points is combined in a Gaussian process. Based on the Gaussian process, the next data point has to be determined. What an optimal next data point is however also depends on the optimization purpose. Acquisition functions aim to find the best "educated guess" by interpreting the knowledge from the Gaussian process toward the optimization objective.
This typically involves choosing between further exploiting regions of the parameter space where the best-known maximum has been found or exploring areas with sparse data. Exploring under-sampled regions targets areas of high uncertainty, where the objective function remains largely unknown.

This issue is known as the "exploration vs. exploitation" dilemma. It is also represented in the example in Fig. 3.1. Specifically, in (B), the algorithm decides to further exploit the area close to the maximal known data points and ends up in a local maximum. Only after the local maximum is further exploited, as represented in (C), does it explore the area that later turns out to contain the global maximum, as represented in (D).
This behavior is a direct consequence of the fact that, in Fig.3.1, each new data point is determined by the maximum of the confidence interval, as outlined in Sec. 3.2. This approach is more than sufficient if the optimization objective is, for example, to find only any data point with an objective value ≥ 1. However, this strategy is less suitable if the optimization goal is to find the global maximum, regardless of the absolute objective value.
This short excursion outlines that the preferences of optimization need to be considered in the decision process of a new observation. This is where acquisition functions come into play. An acquisition function α scores the points of the domain based on the optimization preferences and under consideration of the posterior distribution.

$$\alpha(\mathbf{x}; p_{\text{post}}) : \mathcal{D} \rightarrow \mathbb{R} \tag{3.16}$$

Compared to the expensive-to-evaluate objective function, the acquisition function must be fast to evaluate and analytically differentiable. The following will provide an overview of the three most common acquisition functions: the upper confidence bound in Sec.3.3.1, the probability of improvement in Sec.3.3.2, and the expected improvement in Sec. 3.3.3. The sections are based on the principles and methods discussed in the works of Garnett [30], Shahriari et al. [32], and Wilson et al. [33].

3.3.1 Upper Confidence Bound (UCB)

The Upper Confidence Bound Acquisition function α_{UCB} is the most common representative of the optimistic optimization policies. This approach treats uncertainty as a potential reward and selects the next data point based on the best-case scenario within the assumed confidence interval. Mathematically, this can be achieved by adding the posterior mean prediction μ to the posterior standard deviation σ. These two summands can be manually scaled by the hyperparameter κ. A high κ gives more weight to the uncertainty and encourages exploration, while a low κ enhances

exploitation in the areas where the mean prediction is higher.

$$\alpha_{\text{UCB}}(x; p_{\text{post}}) = \mu(x) + \kappa\sigma(x) \tag{3.17}$$

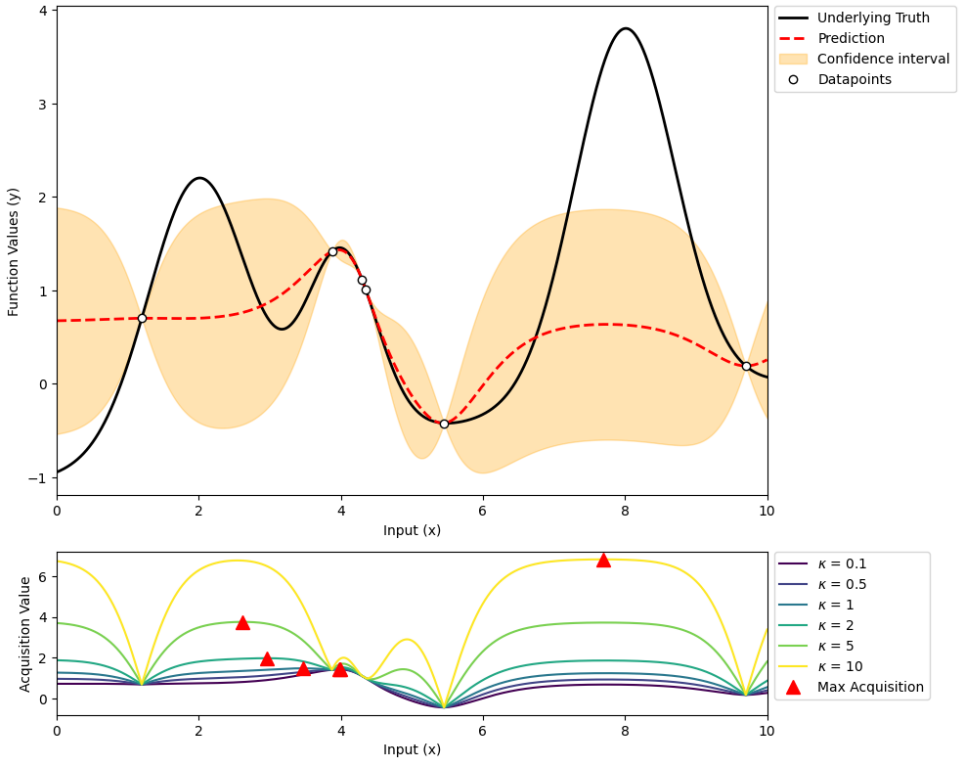


FIGURE 3.2: UCB Acquisition Function for Bayesian Optimization. The exploration-exploitation parameter κ is defined in Eq. 3.17. Top Plot: The black line represents the objective function as the "underlying truth." The red line depicts the current estimate of the objective function, while the orange shaded region represents the confidence interval of the GP. Bottom Plot: Acquisition functions for different values of κ, demonstrating how varying κ influences the exploration-exploitation trade-off. The red triangles indicate the maximum points of the acquisition functions, representing the next suggested query points by the optimizer.

The choice of κ is crucial for efficient optimization. Fig. 3.2 displays the influence of κ on the next evaluated data point. The top plot shows a GP with the prediction and the confidence interval for 6 data points referring to an underlying truth. The bottom plot displays the UCB acquisition functions with varying values for κ. Depending on the κ value, different maxima will be explored in the next iteration of optimization. $\kappa = 0.1 - 5$ will focus on the two local maxima at $x = 2$ and $x = 4$, while only $\kappa = 10$ focuses on the global maximum at $x = 8$. Therefore, a greedy approach would be to choose $\kappa = 10$ to achieve the fastest convergence. However, Tab. 3.1 shows that this intuitive approach is not always optimal. Without the 6 data points provided in Fig. 3.2, $\kappa = 2$ converges the fastest. Convergence, in this case, is considered to be reached when a data point is close to the global maximum, $|x_{\text{best}} - 8| \leq 0.05$. This example of the exploration-exploitation dilemma highlights the importance of hyperparameter tuning for the UCB acquisition function.

TABLE 3.1: Number of iterations until convergence with varying values of the exploration-exploitation parameter κ in the UCB acquisition function. The objective function is expressed in Fig. 3.2. Convergence is assumed to be reached at $|x_{\text{best}} - 8| \leq 0.05$.

κ	0.5	1	2	5	10
Iterations until convergence	36	11	8	12	17

3.3.2 Probability of Improvement (PI)

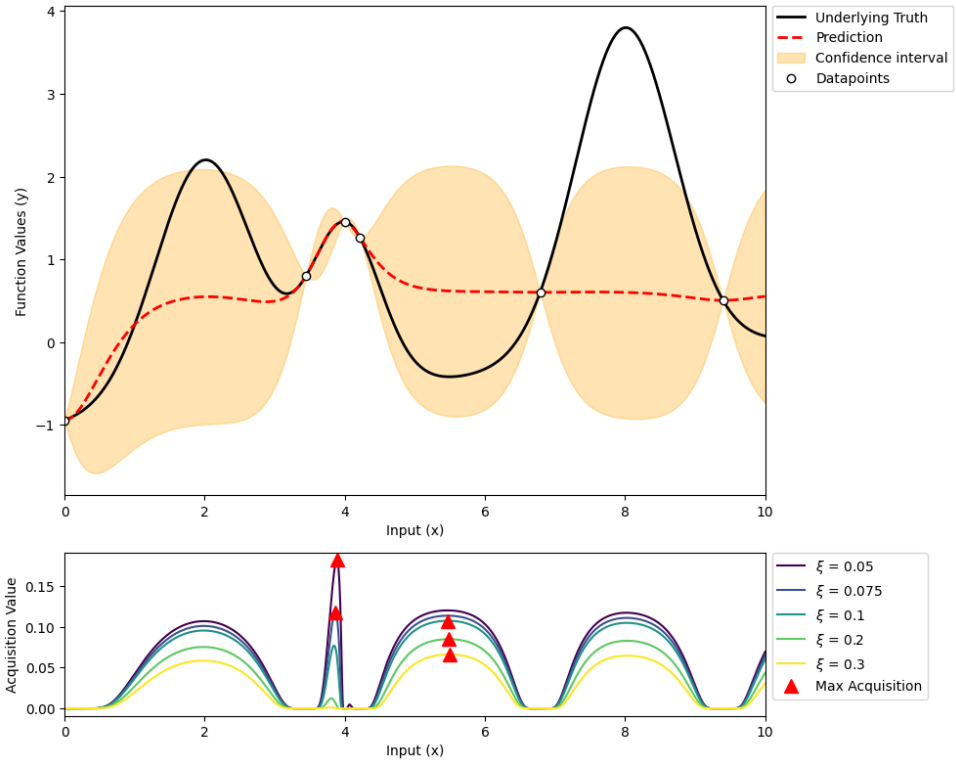


FIGURE 3.3: PI Acquisition Function for Bayesian Optimization. The exploration-exploitation parameter ξ is defined in Eq. 3.19. Top Plot: The black line represents the objective function as the "underlying truth." The red line represents the current estimate of the objective function, while the orange shaded region represents the confidence interval of the GP. Bottom Plot: Acquisition functions for different values of ξ, demonstrating how varying ξ influences the exploration-exploitation trade-off. The red triangles indicate the maximum points of the acquisition functions, representing the next suggested query points by the optimizer.

The Probability of Improvement Acquisition Function α_{PI} belongs to the so-called improvement-based optimization policies. It approximates the likelihood of improvement over a given threshold. Therefore, it does not consider the magnitude of the improvement but only the probability that the next data point will improve compared to the given threshold. This threshold τ is usually the best observed value, $f(x_{\text{best}})$.

$$\alpha_{\text{PI}}(x; p_{\text{post}}) := \mathbb{P}(f(x) > \tau) \tag{3.18}$$

In practice, α_{PI} tends to avoid high-risk decisions, as it prefers a parameter region with a certain moderate improvement over a parameter region with an uncertain improvement of high magnitude. Therefore, it tends to exploit rather than explore. To enable manual adjustment of the exploration/exploitation tradeoff, the parameter ξ is added. With greater ξ values, exploration is enhanced.

$$\alpha_{\text{PI}}(x; p_{\text{post}}) = \mathbb{P}(f(x) > \tau + \xi) \tag{3.19}$$

Fig. 3.3 showcases the characteristics of α_{PI}. First of all, it is noticeable that, since $\tau = f(x_{\text{best}})$, no x values are considered whose prediction is lower than the known optimum. As a result these acquisition values are set to zero. Secondly, Fig. 3.3 presents the influence of the hyperparameter ξ. For low ξ values, the maximum of α_{PI} is close to the local maximum at $x = 4$, where the maximal expected improvement is lower than in other areas with fewer data points but a higher magnitude of uncertainty. This is due to the setup of α_{PI}, which prefers areas with a high probability of improvement over areas with a high magnitude of improvement. Higher ξ values correspond to more exploration. Tab. 3.2 shows the impact of hyperparameter tuning, where the number of iterations until convergence is presented for different ξ values. Low ξ values, in the presented case, spend too many iterations exploring local maxima instead of finding the global one.

TABLE 3.2: Number of Iterations Until Convergence with varying values of the exploration-exploitation parameter ξ in the PI acquisition function. The objective function is expressed in Fig. 3.3. Convergence is assumed to be reached at $|x_{\text{best}} - 8| \leq 0.05$.

ξ	0.05	0.075	0.1	0.2	0.3
Iterations until convergence	25	16	11	12	9

3.3.3 Expected Improvement (EI)

As the name suggests, the Expected Improvement (EI) function is still an improvement-based acquisition function, but in comparison to the Probability of Improvement (PI), it focuses on the expected magnitude of the improvement. α_{EI} prioritizes regions with the potential for improvement over regions with a high probability of improvement.

$$\alpha_{\text{EI}}(x; p_{\text{post}}) := \mathbb{E}\left[(f(x) - \tau)\mathbb{I}(f(x) > \tau)\right] \tag{3.20}$$

To accomplish this objective, Eq. 3.20 makes use of the improvement function $\mathbb{I}$, which is greater than 0 only if $f(x) > \tau$. EI is therefore naturally more inclined towards exploration since unknown areas are more likely to have a higher potential for improvement. Fig. 3.4 shows how the parameter ξ can be used to enforce exploitation.

$$\alpha_{\text{EI}}(x; p_{\text{post}}) := \mathbb{E}\left[(f(x) - \tau)\mathbb{I}(f(x) > \tau + \xi)\right] \tag{3.21}$$

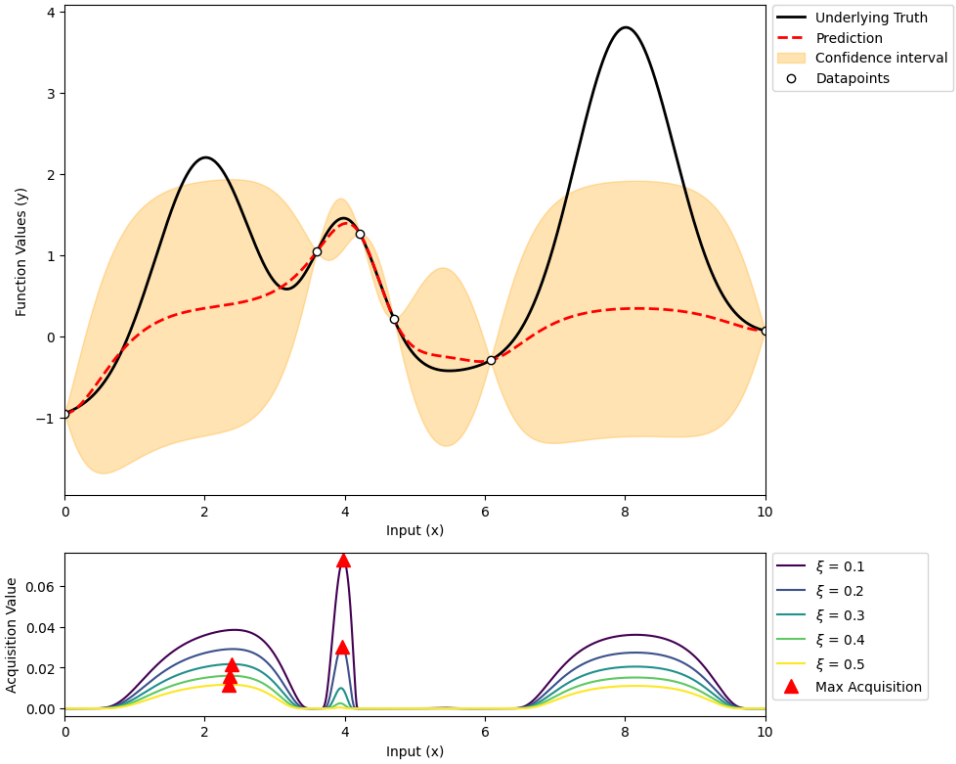


FIGURE 3.4: EI Acquisition Function for Bayesian Optimization. The exploration-exploitation parameter ξ is defined in Eq. 3.21. Top Plot: The black line represents the objective function as the "underlying truth." The red line represents the current estimate of the objective function, while the orange shaded region represents the confidence interval of the GP. Bottom Plot: Acquisition functions for different values of ξ, demonstrating how varying ξ influences the exploration-exploitation trade-off. The red triangles indicate the maximum points of the acquisition functions, representing the next suggested query points by the optimizer.

Fig. 3.4 illustrates the behavior of the α_{EI} acquisition function. According to Eq. 3.20, α_{EI} also does not consider any x with $f(x) < f(x_{\text{best}})$. In general, it does not consider x values that are close to any data points because the probability of improvement is not high enough. Consequently, α_{EI} tends to favor exploration. This is demonstrated in Fig. 3.4, where, even though a local maximum is known at approximately $x = 4$, it is not necessarily selected since other areas present a higher potential for improvement and are therefore explored. However, α_{PI} can be forced to exploit the local maximum with a low ξ. Tab. 3.3 highlights this behavior, demonstrating the necessity of $\xi \leq 0.25$ to efficiently enforce exploitation and the high sensitivity to hyperparameter tuning.

TABLE 3.3: Number of Iterations Until Convergence with varying values of the exploration-exploitation parameter ξ in the EI acquisition function. The objective function is expressed in Fig. 3.4. Convergence is assumed to be reached at $|x_{\text{best}} - 8| \leq 0.05$.

ξ	0	0.05	0.1	0.25	0.5
Iterations until convergence	10	9	6	138	138

3.4 Convergence and Stopping Criteria

In optimization, convergence is typically assessed based on the improvement across iterations. Due to the statistical nature of Bayesian Optimization (BO) and its inherent switching between exploration and exploitation, the improvement achieved in each iteration can vary widely. Consequently, specific stopping criteria must be defined for BO.
BO is designed around the concept of runtime-efficient optimization of expensive-to-evaluate objective functions. Therefore, a predetermined runtime budget can be used. Once this budget is exceeded, the optimization will be terminated. This approach, however, neglects some of the key advantages of BO. If the budget is too low, the optimization will return an unnecessarily inaccurate result, and if the budget is too high, it will waste computational resources even if further optimization is unlikely. To achieve a more improvement-based stopping criterion, the value of the acquisition function can be used, and the optimization is terminated if the acquisition function drops below a certain threshold. As displayed in Fig. 3.2 - 3.4, these acquisition values can vary by the hyperparameter and depend on the dataset, which changes with each iteration [30].
These issues are addressed with a regret-based termination criterion, where the aim is to estimate the global optimum and continue the optimization process until the actual result reaches the desired level of accuracy [34]. This approach enables a dynamic criterion that adjusts in relation to the estimated global optimum.

Chapter 4

Simulation Framework

As outlined in the Motivation, this work aims to develop a simulation framework for optimizing the shape and size of the free layer of a magnetic TMR sensor. This section outlines the Simulation framework needed for this optimization.
The objective of the optimization is kept flexible and therefore can be adjusted to different use cases. The requirements of a search objective are stated in Sec. 4.1.4, while an example objective is described in Sec. 5.3.1.
The optimization is performed in two stages. The first stage focuses on optimizing the shape, i.e., introducing or reducing edges. As discussed in Sec. 1.2.4, different shapes influence the magnetic behavior due to variations in demagnetizing fields. Therefore, different shapes can be configured in the simulation framework.
The second stage focuses on optimizing the size, referring to the physical dimensions (e.g., width, height, depth) of a given aspect ratio. As illustrated in Fig. 1.3, the aspect ratio of a given shape also affect its magnetic characteristics. Therefore, a continuous optimization routine is applied to determine the optimal size for a fixed shape. This process consists of five main steps, outlined in the following:.

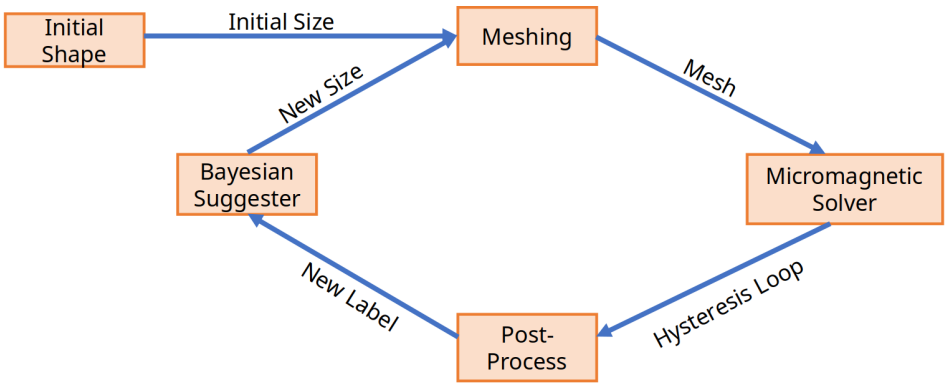


FIGURE 4.1: Schematic workflow of the size optimization of a given shape. Each element is further described in Sec. 4.1.

After defining a specific initial shape, the continuous size optimization process is started. The initial geometry is passed to the meshing procedure, where it is discretized into fine computational elements. On the mesh, the micromagnetic solver performs energy minimization and calculates the equilibrium states for different external field strengths. Combined, these form the hysteresis curve, which is then analyzed in the postprocessing. Once the label is determined in the postprocessing, the label, along with the shape and its specific size, is passed to the optimizer. The

optimizer uses the concept of Bayesian optimization to suggest the next size with different dimensions, which starts the cycle again. This continuous optimization can run until the label determined from the postprocessing meets any convergence criterion or until the computational resources are exhausted. As soon as the optimal size for a given shape is determined, different shapes can be compared to find the optimal sample for the optimization purpose.

Consequently, this work presents the continuous optimization of sizes for discrete shapes. A dynamic shape adjustment, e.g., from a rectangular prism to an elliptical cylinder, is not possible. However, both of these shapes can be configured, and once the optimal size is determined, they can be compared and evaluated.
With mesh sizes surpassing 1 000 000 points, allowing full freedom in point positioning would result in a computationally prohibitive parameter space. This theoretical limitation aligns with real-world constraints, as the growth process of TMR sensor free layers inherently restricts the range of feasible geometries to discrete configurations.

To eliminate any ambiguity between the Bayesian optimizer and the overall optimization framework, the term Bayesian suggester will be used, underlining its purpose of recommending new sizes.

4.1 Components of the Framework

4.1.1 Initial shape

The opportunities for configuring an initial shape are wide. As the initial shape, a script has to be configured, which defines the shape and enables the meshing. In addition, the size parameters (e.g., `"xlen": float, "ylen": float, "zlen": float`) must be specified, along with their corresponding optimization intervals (see Appendix C). These size parameters form the parameter space of the optimization. The Bayesian suggester will attempt to find the size that maximizes the search objective. Any shape that can be continuously meshed with different parameters can be configured with the required size parameters and passed to the optimization cycle.

4.1.2 Meshing

In meshing, any given shape with any given size can be discretized. The shape specifications are defined in the initial shape, and the size parameters are passed in each iteration cycle. Theoretically, any meshing algorithm can be configured. The expected output is a *mesh.unv* file containing a tetrahedral finite element mesh.
In this work, the Salome Simulation Framework [35] is used, configured to create the geometry and to apply the NETGEN [36] meshing algorithm for discretization. An important parameter is the finite element size parameter ℓ_{mesh}, which characterizes the maximal edge length of a finite element. From a software point of view, this parameter can be chosen freely. However, if it is chosen too large, the discretization no longer accurately simulates the physical behavior.

4.1.3 Micromagnetic Simulation

The micromagnetic solver receives a *mesh.unv* file, which defines the finite element discretization of the sample. Following energy minimization, it outputs a *hysteresis.dat* file containing the magnetization components projected onto the direction of the external magnetic field. In this work, the B4vex [22] micromagnetic solver is used to perform energy minimization. It is configured to perform a preconditioned gradient descent as described in Sec. 1.4. As suggested in [22], an approximated Hessian is used for preconditioning.

As described in Sec. 1.1, soft magnets have a remanence close to 0. Therefore, the magnetization configurations are calculated for an external field which is applied along a fixed spatial direction and iteratively decreased. Thus, the resulting curve is not a full hysteresis loop and will hereafter be referred to as a magnetization curve. The parameters specific to the B4vex software are independent of the size and shape of the geometry. As such, they only need to be defined once and remain constant throughout all iterations of the optimization process. Below, an overview of the most important parameters is presented:

Parameter	**Description**	**Values in This Work**
[mx, my, mz]	External magnetic field direction	$[0, 1, 0]$
[hstart, hfinal, hstep]	Start and final values of external magnetic field and the iteration steps	$[0.5, -0.5, 0.001]$ (T)
cg_method	Preconditioned Conjugate Gradient decent according to [22]	1004
precond_iter	Iterations to approx. Preconditioner	10
$\mu_0 M_s$	Saturation magnetization	1 (T)
A	Exchange stiffness constant	1.0×10^{-11} (J/m)

TABLE 4.1: Example B4vex settings for $Ni_{80}Fe_{20}$ soft magnetic thin films.

These parameter are initially configured and provided the B4vex simulation in *material.krn* and *simulation.p*2.

4.1.4 Postprocessing

Postprocessing refers to the steps taken after simulation data is produced, aimed at extracting meaningful physical labels for optimization. The magnetization curve is received from the micromagnetic solver, and a one-dimensional float is calculated as a label for the search objective.
The aim of this work is to enable a broad range of users to optimize toward individual search objectives. Therefore, an 'easy' implementation of individual search objectives is fundamental. A modular, object-oriented approach was chosen to ensure adaptability.
The desired search objective can be implemented as a postprocess. The abstract class *AbstractPostProc* acts as a structural template, specifying the necessary interfaces

that all postprocessing modules must implement within the simulation framework. Therefore, the desired optimization target can be realized by implementing the four methods described in Lis. 4.1. Implementation of these four methods ensures that the simulation framework remains intact while allowing optimization for different search objectives.

In the following, a detailed description of the implementation of a postprocess is outlined.
The initialization of a postprocess requires two entities. First of all, a simulation config, which holds the details about the simulation (Sec. 4.3), and the shape, which is configured in the initial shape setup and adjusted for the specific size of the iteration cycle (Sec. 4.1.1). The four methods of a postprocess require the following implementation:
The first method, *load_file*, receives the location of the magnetization curve in *hysteresis.dat* from the micromagnetic solver as a string and loads the data.
The second method, *calc_label*, calculates the label. This methods forms the core of every postprocessing operation. In theory, every aspect of the magnetization curve could be considered, as well as the size parameters of the sample. Labels such as the minimum slope of the magnetization curve or a defined interval for linear behavior can be implemented without requiring advanced programming experience. Incorporating manufacturing cost enables the definition of labels like the lowest-cost shape that satisfies a minimum slope criterion. The third and fourth methods (*plot_postProc* and *save_postProc_plot*) are meant to display and save an intuitive visualization of the postprocess to enable better traceability and fast verification.

```
class AbstractPostProc(ABC):
    def __init__(self, config: OptimizerConfig, shape: Shape):
        ...
    @abstractmethod
    def load_file(self, location: String):
        pass
    @abstractmethod
    def calc_label(self) -> float::
        pass
    @abstractmethod
    def plot_postProc(self):
        pass
    @abstractmethod
    def save_postProc_plot(self, location: String):
        pass
```

LISTING 4.1: Pseudocode for Abstract Post-Processing Class

The implementation of the example search objective described in Sec. 5.3.1 can be used as a template for the implementation of individual search objectives.

4.1.5 Bayesian Suggester

The aim of the Bayesian suggester is to receive the label associated with the size of a shape from the postprocesses and suggest the next size, which starts the optimization cycle again.
In general, any black-box optimizer could be integrated for this task. In this study, a Bayesian suggester [37] is used, which performs optimization as described in Chapter 3. The code for this software package is open source and available at [38].

It calculates a probability distribution of the entire parameter space, taking into account the labels and configuration of every iteration. Based on this probability distribution, several acquisition functions can be used to suggest the optimal next size of a shape. Within the used package, individual acquisition functions can be configured. In this study, three of the pre-implemented acquisition functions (UCB, PI, EI) were used. As described in Sec. 3.3.1 - 3.3.3, each can be hyperparameter-tuned with an exploration-exploitation trade-off parameter. The acquisition function, as well as its hyperparameter, have to be configured initially and do not change during the iterations.

Again, the object-oriented design of this work enhances the integration of different optimizing tools. Similar to the *AbstractPostProc* class in Sec. 4.1.4, a new instance of the *Optimizer* class can be implemented, which defines the interface to any possible black-box optimizers.

4.2 Parallelization

The minimization of the free energy in a micromagnetic sample, as described in Sec. 1.3, involves an iterative, self-consistent feedback loop where each mesh cell influences all the others, which inherently challenges parallelization. Additional, the meshing of the sample is also usually performed on a single CPU thread.
Despite the use of GPUs for the micromagnetic energy minimization, the time to solution may exceed several hours. In order to achieve a speed-up, several iterations can be computed simultaneously, thereby analyzing different sizes of a shape at the same time. To enable this parallelization, the Bayesian suggester of several optimization circuits are connected via a database. Fig. 4.2 shows the example setup of four optimization circuits, which can be configured on different nodes. Each Bayesian suggester, however, is connected through a global SQL database. Once a new label for a specific size is calculated, it is saved to the database, and before a new size is suggested, the Bayesian suggester loads all other data points and labels from the database.

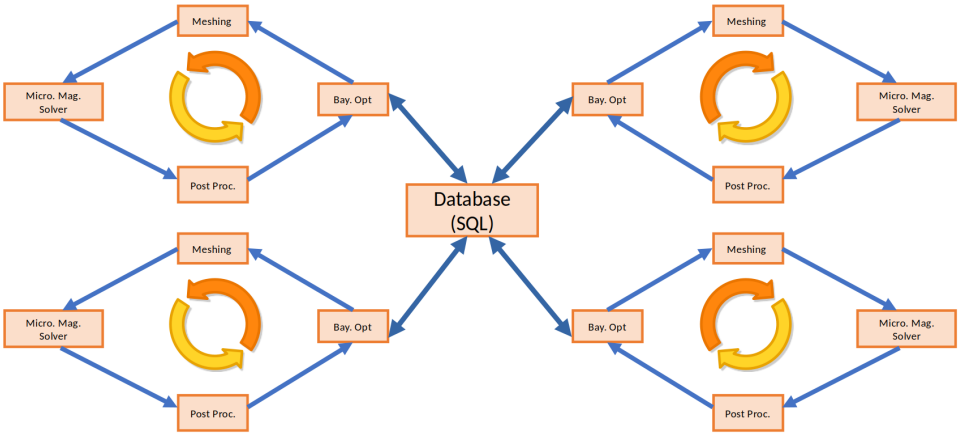


FIGURE 4.2: Schematic workflow of parallelization.

This configuration enables the iteration circuits to work independently from each other while still profiting from the results of all the other circuits. They also do not

have to be synchronized. Since each circuit calculates a different size, the computation times are different. Every time a Bayesian optimizer suggests a new size, it only considers completed iterations from other circuits and is neither aware of how many other optimization circuits are running nor of their status.
This parallelization setup also enables advantages arising from different acquisition functions. As stated in Sec. 3.3, different acquisition functions exhibit different properties or can be hyperparameter tuned to different characteristics. The setup in Fig. 4.2 enables the use of different acquisition functions. One approach may involve setting up an iteration circuit with an exploration-focused acquisition function, alongside another circuit configured for exploitation.

4.3 Configuration

An easy approach to the simulation framework is one of the main goals of this work. To enable users to intuitively use the provided software, all the specific configuration files are collected in a single config file. An example config is provided and described in Appendix C. It is separated into 6 categories.
Under *database*, settings like the global path to the database and a folder to store figures of the postprocesses are configured. With the $use\_DB : False$ configuration, the framework can also be used standalone without parallelization.
In the shape section, the general shapes like a cylindrical ellipse or a rectangular prism can be configured, as well as their initial sizes. An optimization setup for three parameters has been configured. These can, however, be adjusted to the specific shapes as well.
In the *simulation* section, the maximum iteration number as well as meshing parameters can be configured. Also, the parameter intervals of the optimization and the field range, as well as the step length of the magnetization curves, can be set.
Specifications of the Bayesian suggester are configured under *optimizer*. A choice of acquisition function as well as their hyperparameters are provided.
The hardware settings of the server can be configured in the "server" section, and the log level in the *general_setting*.

4.4 Accessibility

This thesis contributes to the MaMMoS research project [1]. The findings of MaMMoS will be open source and therefore publicly available after the project has finished. The repository is documented, and the configuration files enable a low entry threshold for applications to the already implemented functions. Additional functions, such as other search objectives or optimizers, can be extended through the object-oriented structure.

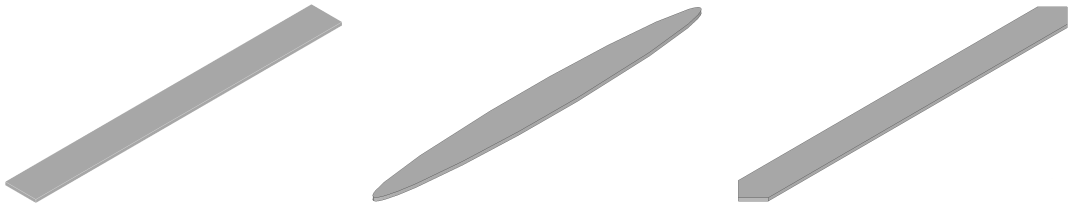
Chapter 5

Shape Optimization

This section is intended to present a use case of the simulation framework described in Chapter 4. To demonstrate the capabilities of the simulation framework, the three representative shapes shown in Fig. 5.1 are analyzed. Fig. 5.1A shows a rectangular prism, Fig 5.1B, an elliptical cylinder, and Fig 5.1C, a rectangular prism with tapered ends. All three shapes are characterized by three size factors: x_{len}, y_{len}, z_{len}.
For the rectangular prism, these simply describe the expansion along the three spatial dimensions. In the case of the elliptical cylinder, x_{len} and y_{len} describe the major and minor axes, while z_{len} defines the height of the cylinder. The rectangular prism with tapered ends is defined as a rectangular prism without tapered ends, and then they are added at an angle of 90°. The length of the tapered ends is therefore not individually adjustable and not included in x_{len}.
These three size parameters will be optimized for every shape in the same parameter interval given in Tab. 5.1.

Parameter	Description	Interval (nm)
x_{len}	Length in x-direction	$[1000, 2000]$
y_{len}	Length in y-direction	$[100, 200]$
z_{len}	Length in z-direction	$[10, 20]$

TABLE 5.1: Parameter intervals in the x, y, and z directions for the three shapes illustrated in Fig. 5.1.



(A) Rectangular Prism (B) Elliptical Cylinder (C) Rectangular Prism with Tapered Ends

FIGURE 5.1: Comparison of three shape configurations: (A) Rectangular Prism, (B) Elliptical Cylinder, (C) Rectangular Prism with Tapered Ends.

5.1 Magnetization Curve

As described in Sec. 1.3, the micromagnetic solver calculates the equilibrium state according to the free energy for a given external field. In this chapter, the external field is always applied along the y-axis. If this external field is reduced over a sufficient interval, a numerical approximation of a magnetization curve is calculated. Example magnetization curves for the configuration $x_{\text{len}} = 2000$ nm, $y_{\text{len}} = 100$ nm, $z_{\text{len}} = 20$ nm of the three shapes are displayed in Fig. 5.2.

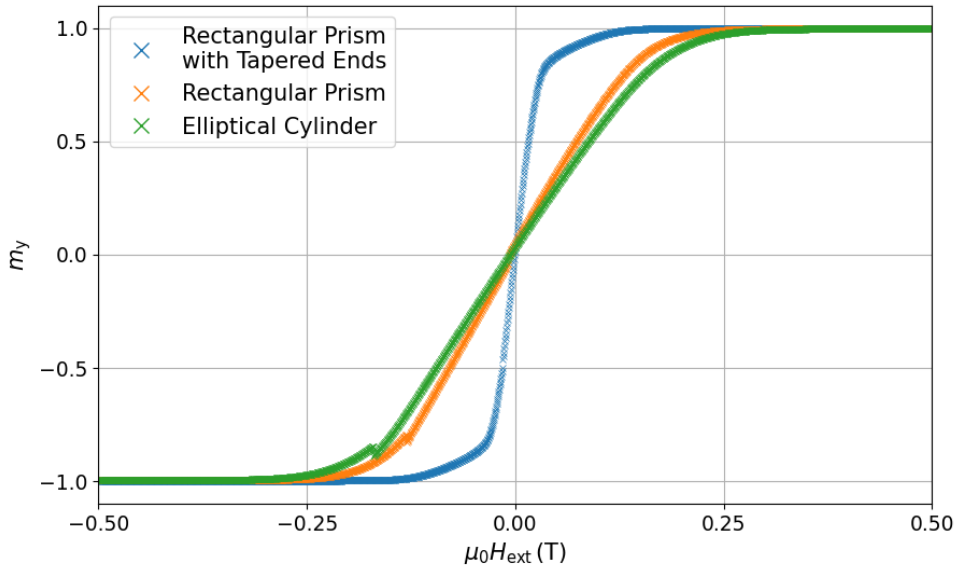


FIGURE 5.2: Combined numerically approximated magnetization curves for different shapes. Each shape is configured with the size parameters of $x_{\text{len}} = 2000\,\text{nm}$, $y_{\text{len}} = 100\,\text{nm}$, $z_{\text{len}} = 20\,\text{nm}$. The external field is applied along the y-axis.

Fig. 5.2 provides several insights. First, the micromagnetic energy minimization yields the magnetization curves as expected in Sec. 1.1. The magnetization curves are generated by sweeping the external magnetic field from 0.5 T to -0.5 T, beginning and ending in saturation. Since the field is applied along the hard axis, the remanence is negligible, making a reverse sweep redundant. The material's current magnetic state is largely unaffected by its previous magnetization states. Artifacts introduced during the meshing process are, for example, one source of deviations between numerical and analytical magnetization curves. These will be examined in Sec. 5.2.
Secondly, it is evident that the different shapes also show, around $H_{\text{ext}} \approx 0$, a different slope in the magnetization curves, even though the sizes are comparably similar. This underlines the effect of the shape of a sample on its magnetization curve and will be further discussed once the optimal size of each shape is determined in Sec. 5.3.

In Sec. 1.2.4, analytical solutions of the magnetization curves were developed. Fig. 5.3 compares these analytical functions with numerical simulations of the same samples. The analytical solutions are the same as depicted in Fig. 1.5, and the three samples are described in Tab. 1.1. They are all rectangular prisms with $z_{\text{len}} = 20\,\text{nm}$ and

three different aspect ratios ($\alpha_{xy} = x_{\text{len}}/y_{\text{len}}$) in the xy-plane. Rectangular prisms with the corresponding aspect ratios are also illustrated in Fig. 5.3.

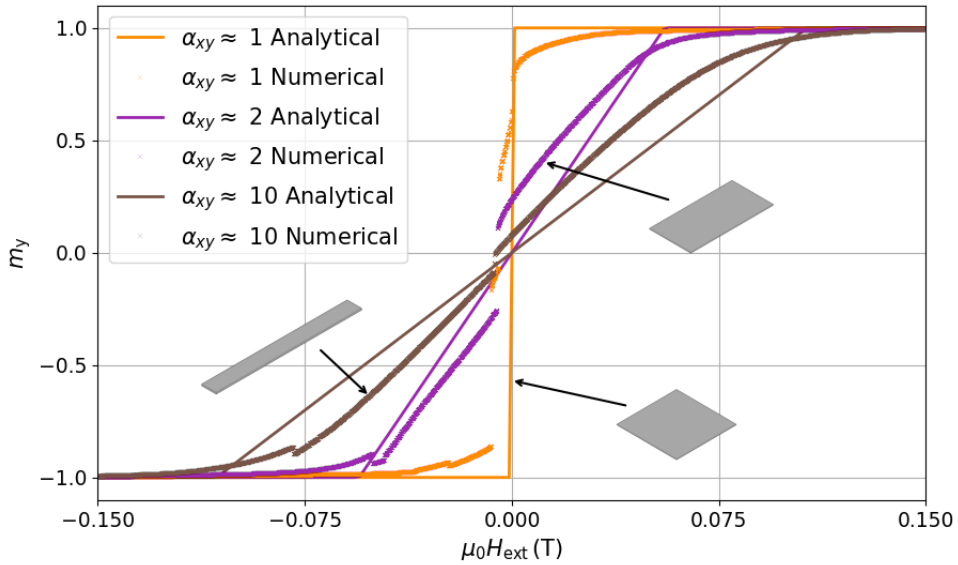


FIGURE 5.3: Comparison of numerically simulated magnetization curves with analytical predictions for varying aspect ratios. Shown are three rectangular prisms with varying aspect ratios; their geometries are sketched in gray and listed in Tab. 1.1.

The theoretical assumptions of homogeneous magnetization and homogeneous demagnetization fields are rather strict and will, in practice, never be fully met especially not by a rectangular prism such as those in Fig. 5.3. Therefore, discrepancies between the numerical and analytical solutions are to be expected. Nevertheless, they do show similar behavior. Both the numerical and analytical solutions demonstrate that rectangular prisms with a higher aspect ratio also exhibit a reduced slope in the magnetization curve. This observation is currently based on only three data points but will be further supported by the findings in Sec. 5.3. Fig. 5.3 also shows significant kinks in the magnetization curves. They will be further discussed in the next section.

5.2 Kinks in the Magnetization Curve

Fig. 5.2 and Fig. 5.3 show distinct kinks in their magnetization curves. Each curve exhibits these features, although their magnitude varies depending on the specific shape and size. The kinks may arise from various sources, some of which can be explained by examining the magnetic configuration. In Fig. 5.4, the magnetization curve of a rectangular prism with the parameters $[2000, 100, 20]$ (nm) is displayed. The kinks are highlighted in red, and five characteristic states of the magnetization are labeled, with the corresponding magnetization configurations displayed below.

In State (1), the magnetization is homogeneously aligned in the y-direction, parallel to the external field. Once the external field is reduced, the magnetizations align along the easy axis in the direction of the x-axis, as seen in state (2). The preferential alignment of magnetization along the sample boundaries renders the y-oriented end faces conducive to parallel magnetic moment alignment. As a result, State (2)

exhibits the so-called "S-state" configuration. The magnetization on these end faces still points in the positive y-direction, while the remaining magnetizations are already aligned with the x-axis. These different states of soft magnetic thin film elements were previously investigated by Miltat and Donahue [39].

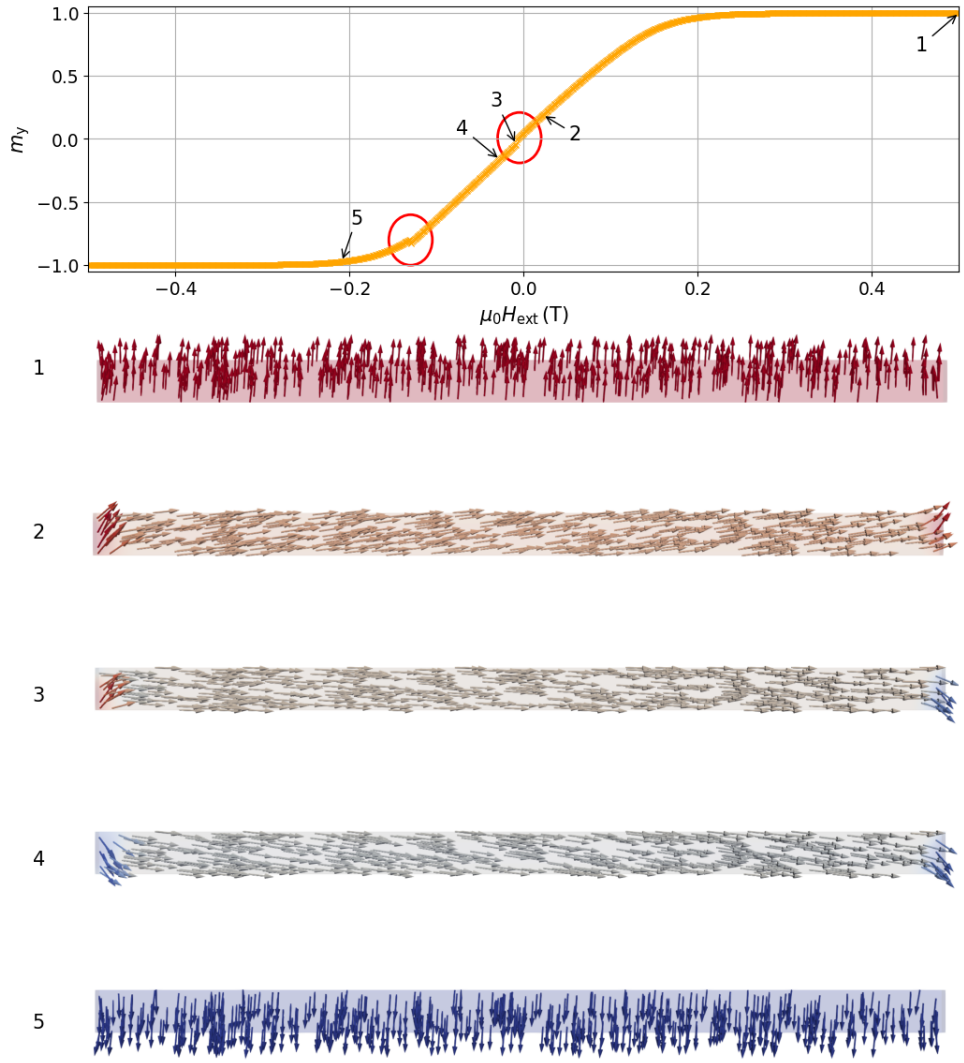


FIGURE 5.4: Magnetization curve and magnetization configurations. (Top) The magnetization curve of an rectangular prism with size parameters $[2000, 100, 20]$ (nm). The external magnetic field H_{ext} applied in y direction, with distinct regimes (1)-(5) marked by arrows. Red ellipses highlight key kinks corresponding to magnetic transitions. (Bottom) The magnetic configuration images, corresponding to regimes (1)–(5).

Once the external magnetic field is further reduced to negative values, a so-called "C-state" is reached in (3); one of the end faces flips its magnetization toward the negative y-direction. This "flip" causes discrete changes in the orientation of the magnetization and can result in kinks in the magnetization curve. In this instance, the kink in the magnetization curve at $H_{\text{ext}} = -0.009\,\text{T}$ is caused by the transition from configuration (2) to (3). Regardless of how gradually the magnetic field is reduced, these transitions can appear as discrete kinks, since alignment with the end faces—either positive or negative—is energetically more favorable than continuous

rotation. For a rectangular prism, this transition from the "S-state" to the "C-state" always appears at negative external field values, as a negative field is required to trigger the reversal process.
As the external field is further reduced, the inverse "S-state" (4) is reached, where the magnetizations on both end faces are oriented in the negative y-direction. However, the transition from (3) to (4) does not result in a kink in the magnetization curve, as the second kink only appears once the external field is further reduced to $H_{\text{ext}} = -0.13\,\text{T}$. State (5) displays saturation again, where the magnetization reaches saturation in the negative y-direction.
These examples highlight the relationship between kinks in the magnetization curve and the magnetic configuration, demonstrating that the origin of each kink must be examined individually.

The magnetic configurations of an elliptical cylinder and a rectangular prism with tapered ends show similar characteristics to the rectangular prism discussed above. However, the different shapes lead to different positions of the kinks in the magnetization curves. These rapid yet continuous transitions can still negatively affect the sensing performance of TMR sensors. The avoidance of kinks can be incorporated into the search objectives of the simulation workflow (Sec. 4.1.4).

5.3 Continuous Optimization

The sections above introduced the three shapes (Fig. 5.1) and their parameter intervals (Tab. 5.1). This section focuses on the continuous size optimization of these shapes using the workflow described in Chapter 4. To enable continuous optimization, a search objective must be developed which evaluates the magnetization curve of a given shape and size enabling comparison across configurations.

5.3.1 Search Objective: Field Range of Linear Response R_{lin}

Tunneling magnetoresistance sensors are designed to measure external magnetic fields. Several characteristics of the magnetization curve can enhance or impair the sensitivity of a free layer. To showcase the capability of the simulation framework, the field range in which the magnetization of the free layer shows a linear response to changes in the external magnetic field R_{lin} is used as a search objective. For computational efficiency and to minimize the influence of kinks, the analysis is restricted to the positive half of the magnetization curve, where $H_{\text{ext}} > 0$.
This search objective aims to maximize the operational range of the sensors. It does not consider that a large field range results in a nearly horizontal magnetization curve, which reduces the sensitivity. This search objective is intended as an example, demonstrating the capabilities of the simulation framework and serving as a guideline for implementing postprocessing requirements as outlined in Sec. 4.1.4.

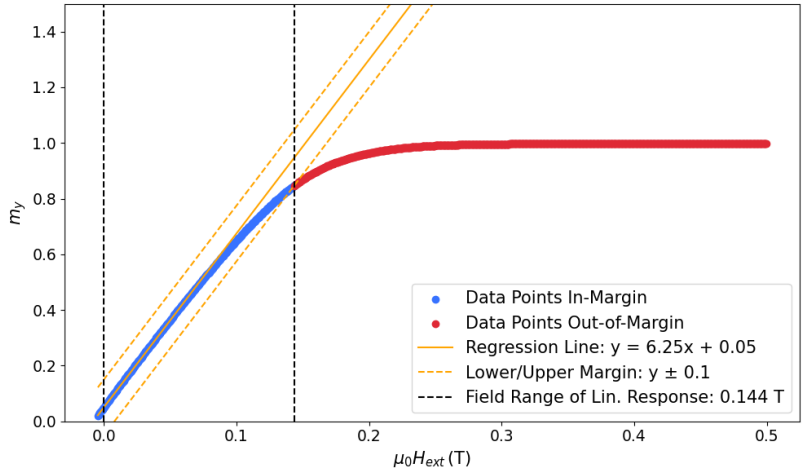


FIGURE 5.5: Magnetization curve for positive external field values and schematic illustration of the search objective. The orange line indicates the linear regression around $H_{\text{ext}} = 0$, with dashed orange lines showing the upper and lower margins. Data points within the margin (indicated in blue) are considered to exhibit linear behavior, whereas those outside the margin (shown in red) deviate from linearity. The black dashed lines mark the estimated field range in which the system's response is approximately linear — this range defines the search objective.

Fig. 5.5 visualizes the field range in which the sample shows a linear response to variations in a positive external magnetic field. The calculation of R_{lin} consists of three steps:

1. A linear regression of the magnetization curve around $H_{\text{ext}} = 0$: an interval around zero is defined and fitted using linear regression. This regression defines the reference for linear behavior. In the following, the interval $[-0.001, 0.05]$ T is used to extract the slope of the magnetization curve.
2. A margin $\epsilon > 0$ is defined, within which the magnetization curve is still considered to be close to the linear regression. The system is still considered to respond linearly to changes in the external field. In practice, $\epsilon = 0.1$ has proven to be a suitable trade-off and will be used in the following. If ϵ is chosen too large, kinks in the magnetization curve (Sec. 5.2) might still be considered as linear behavior. If ϵ is chosen too small, numerical inconsistencies may lead to misinterpretation as non-linear behavior.
3. The distance of linear behavior is then calculated by the x-value of the last data point within the ϵ margin around the linear regression. Accordingly, R_{lin} is discretized by the step size of H_{ext} in the magnetization curve.

5.3.2 Rectangular Prism

Fig. 5.6 displays 1 000 iterations of the optimization framework described in Chapter 4. The optimization targets a rectangular prism within the parameter intervals given in Tab. 5.1 and demonstrates that the simulation framework is successfully optimizing the sizes of a given shape. Each data point in Fig. 5.6 represents one

iteration of the workflow shown in Fig. 4.1. It is visible that the Bayesian optimizer first explores the entire parameter range for about 200 iterations, evaluating a broad range of search objective values. After 200 iterations, the Bayesian optimizer starts exploiting the best known maximum, evaluating almost only data points with $R_{\text{lin}} \approx 1.4$ T. Significant drops in the evaluated label at ≈ 550 and ≈ 750 iterations suggest that the Bayesian optimizer resumes exploration in an attempt to locate another maximum.
However, the maximum of the search objective is already found after 181 iterations, during the exploration phase of the optimization. At the optimal point, given by $[x_{\text{len}}, y_{\text{len}}, z_{\text{len}}] = [1929, 100, 20]$ (nm), the maximum observed value of R_{lin} was 0.144 T.
In this example, the Bayesian optimizer was set up with the UCB acquisition function and an exploration–exploitation parameter of $\kappa = 2.5$. Further discussion of acquisition functions is provided in Sec. 5.4.

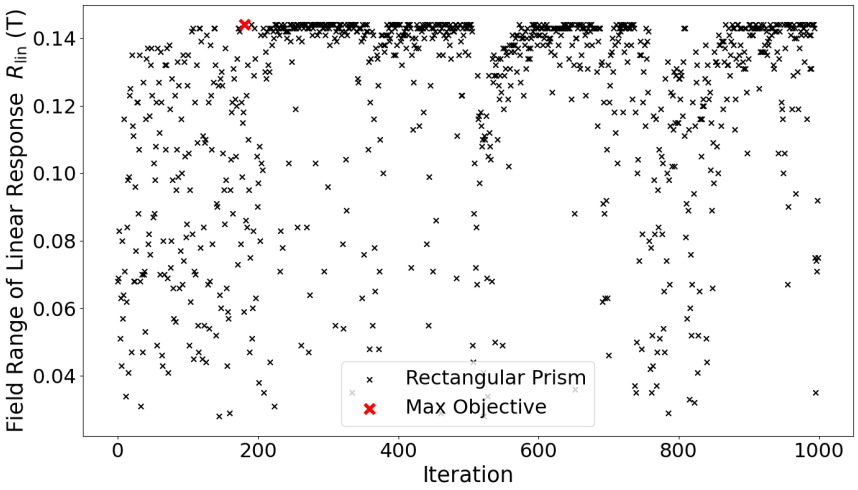


FIGURE 5.6: Continuous size optimization of a rectangular prism within the parameter interval given in Tab. 5.1. Each data point represents one iteration of the simulation framework shown in Fig. 4.1. The iteration with the maximum field range of linear response is shown in red.

5.3.3 Elliptical Cylinder

Fig. 5.7 displays the first 1 000 iterations of the optimization of an elliptical cylinder. The optimization shows similar characteristics to that of the rectangular prism in Sec. 5.3.2. As shown in Fig. 5.7 the Bayesian optimization also takes about 200 iterations to explore the parameter range before exploiting the known maxima. Since both optimizations are performed with a UCB acquisition function and $\kappa = 2.5$, similar behavior is expected. In the case of the elliptical cylinder, the optimization starts exploring again after 350 iterations, indicating that the known maxima might be exhausted. The optimum was found after 185 iterations at $[x_{\text{len}}, y_{\text{len}}, z_{\text{len}}] = [1927, 100, 20]$ (nm), yielding a maximum observed value of $R_{\text{lin}} = 0.172$ T.

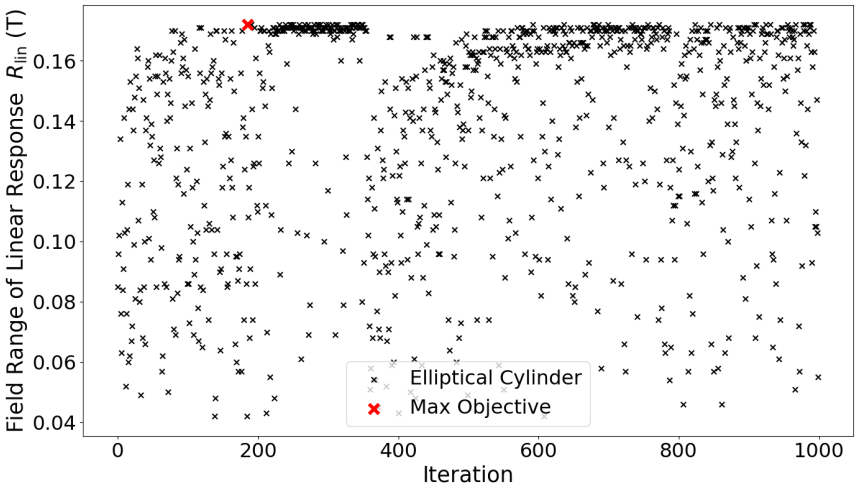


FIGURE 5.7: Continuous size optimization of a elliptical cylinder within the parameter interval given in Tab. 5.1. Each data point represents one iteration of the simulation framework shown in Fig. 4.1. The iteration with the maximum field range of linear response is shown in red.

5.3.4 Rectangular Prism with Tapered Ends

Fig. 5.8 displays the first 630 iterations of the optimization of a rectangular prism with tapered ends. This study considers fewer iterations compared to those of the rectangular prism and the elliptical cylinder, because the numerical energy minimization is more computationally expensive than for the other two shapes. The Bayesian optimization is configured identically to that used for the rectangular prism and the elliptical cylinder, employing a UCB acquisition function with $\kappa = 2.5$.
The optimization of the rectangular prism with tapered ends, however, exhibits different characteristics compared to the other two shapes. At 301 iterations, the estimated optimal size was found to be $[x_{\text{len}}, y_{\text{len}}, z_{\text{len}}] = [1030, 100, 20]$ (nm), achieving a linear response value of $R_{\text{lin}} = 0.0253\,\text{T}$. Accordingly, the optimal x_{len} deviates significantly from the values obtained for the other shapes. Additionally, after the 630 depicted iterations, no clear exploitation phase is observed, and convergence has not necessarily been achieved. This suggests a more complex objective landscape featuring multiple local optima which may hinder convergence. The tapered ends introduce additional inhomogeneity into the demagnetizing field. An inhomogeneous demagnetizing field may give rise to additional local maxima in the optimization landscape. Furthermore, the creation of suitable finite element meshes that keep the computational effort low while maintaining a suitably low number of finite elements is difficult.

Sec. 5.5 suggests that a maximum finite element side length of 5 nm already poses challenges for a plain rectangular prism. The tapered ends introduce additional complexities for the meshing process. Differences in the meshing of the tapered ends, even for similar parameter configurations, could lead to varying results in R_{lin}. If a rectangular prism with tapered ends were to be configured as the free layer of a TMR sensor, this aspect would need to be further investigated. Considering both the hight computation time per iteration (up to 8 h) and the fact that their objective values are an order of magnitude lower than those of the other shapes, they are not

further considered in this study.
Nevertheless, this case demonstrates that more complex shapes can be both configured and automatically optimized using the presented simulation framework.

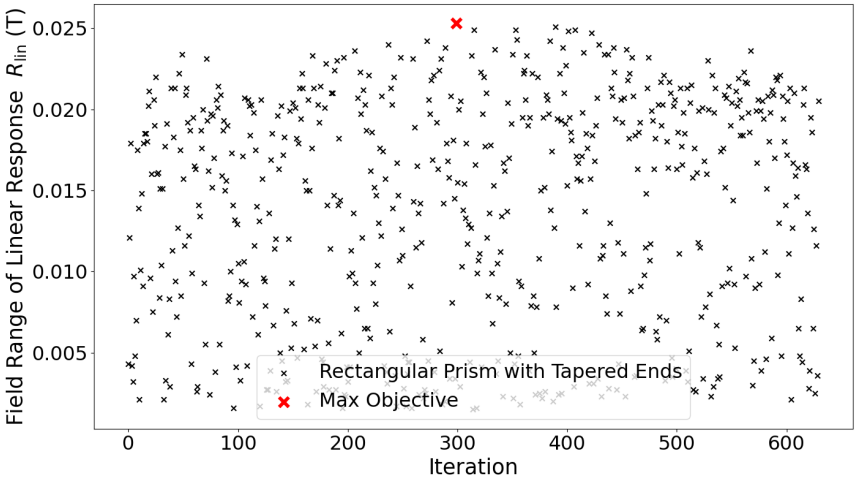


FIGURE 5.8: Continuous size optimization of a rectangular prism with tapered ends within the parameter interval given in Tab. 5.1. Each data point represents one iteration of the simulation framework shown in Fig. 4.1. The iteration with the maximum field range of linear response is shown in red.

5.3.5 Comparison of the Three Shapes

The preceding three sections presented size optimizations for three different shapes. The sizes which show the largest field range of linear behavior are displayed below.

Shape	$[x_{\text{len}}, y_{\text{len}}, z_{\text{len}}]$ (nm)	R_{lin} (T)
Rectangular Prism	[1929, 100, 20]	0.144
Elliptical Cylinder	[1927, 100, 20]	0.172
Rectangular Prism with Tapered Ends	[1030, 100, 20]	0.025

TABLE 5.2: Comparison of the optimal size of the three different shapes from Fig. 1.2. The individual optimizations are discussed in Sec. 5.3.2–5.3.4 and were performed in the parameter interval where $[x_{\text{len}}, y_{\text{len}}, z_{\text{len}}] \in [1000\text{–}2000,\ 100\text{–}200,\ 10\text{–}20]$ (nm).

Tab. 5.2 shows that, among the three observed shapes, an elliptical cylinder is the most suitable shape for a free layer of a TMR sensor. With a field range of linear behavior of 0.172 T, the elliptical cylinder shows promising potential. As described in Sec. 5.3, R_{lin} only considers the positive field range. Since, in practice, the full magnetization curve may contribute to sensing, an even larger field range is expected.
The rectangular prism, both with and without tapered ends, show a reduced field range of linear behavior. These results arise from the behavior of the magnetic moments: in shapes with fewer edges, the magnetic moments exhibit greater rotational freedom within the sample. A rectangular prism with tapered ends has more edges than an elliptical cylinder and therefore imposes more constraints on the magnetic

moments, leading to a smaller field range of linear response. The tapered ends, in particular, seem to significantly influence the field range of linear response.

The optimal sizes of the elliptical cylinder and the rectangular prism are similar, as both exhibit their maxima at $[1928 \pm 1,\ 100,\ 20]\,(\mathrm{nm})$, indicating that for these rather smooth shapes, the size is governed by the same physical phenomenon. As Sec. 5.3.4 suggests, the rectangular prism with tapered ends displays different behavior. With an optimal $x_{\mathrm{len}} = 1030\,\mathrm{nm}$, it lies at the opposite end of the parameter interval.

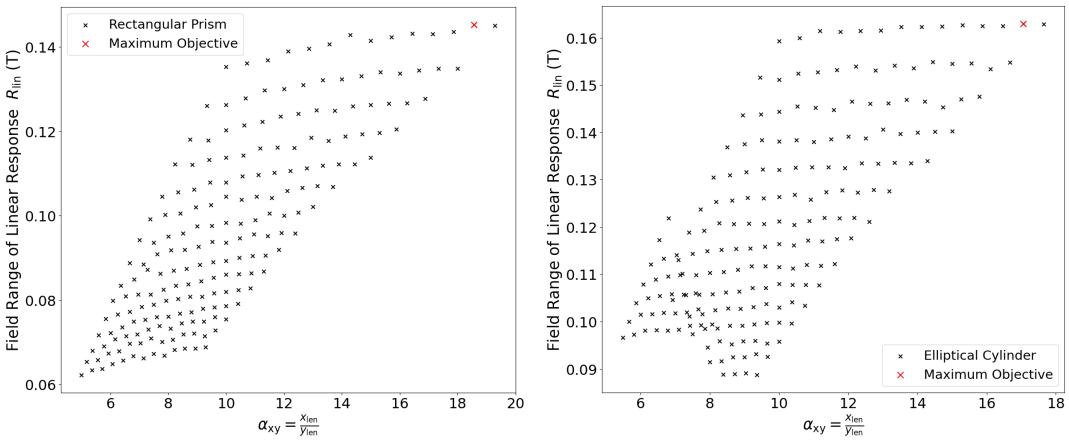
Sec. 1.2.4 concludes that a higher aspect ratio (α_{xy}) of the x_{len} to y_{len} length leads to a reduced slope in the magnetization curve and therefore an increased R_{lin}. The numerical results show that this holds true only for the rectangular prism and the elliptical cylinder. The following section further examines the relationship between the theoretical findings and the numerical ones.

5.3.6 Comparison of Analytical and Numerical R_{lin}

To compare the numerical findings with the analytical predictions in Sec. 1.2.4, different sizes with the same z-length but varying xy-dimensions were considered. At a fixed z-length of 20 nm, a grid of 200 uniformly spaced configurations in the xy-plane was sampled.

$$[x_{\mathrm{len}}, y_{\mathrm{len}}, z_{\mathrm{len}}] \in [1000\text{–}2000,\ 100\text{–}200,\ 20]\,(\mathrm{nm}) \tag{5.1}$$

The numerical simulations of R_{lin} for the given parameter space are shown in Fig. 5.9.



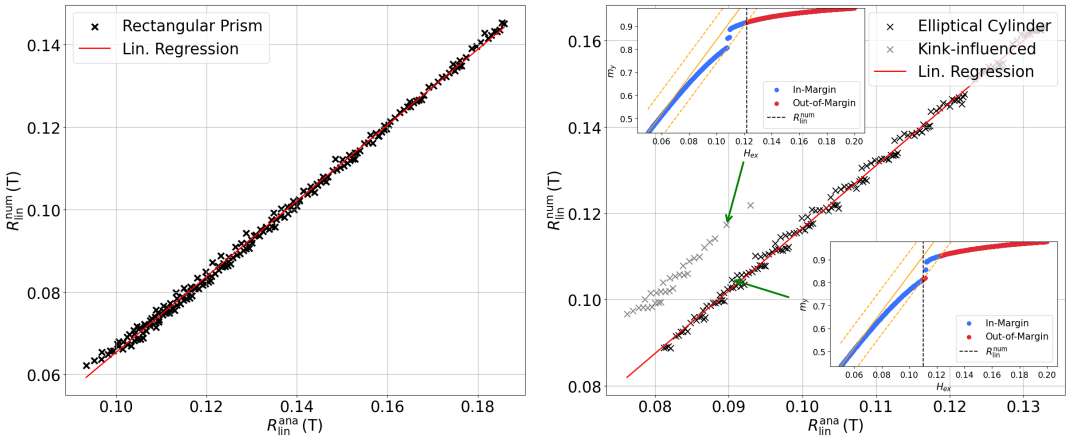
(A) Rectangular Prism

(B) Elliptical Cylinder

FIGURE 5.9: Numerical simulations of the field range of linear response R_{lin} for a rectangular prism (A) and an elliptical cylinder (B). Displayed are 200 uniformly spaced sizes in the parameter range in Eq. 5.1. The data point corresponding to the maximum objective value is highlighted in red.

Both the rectangular prism and the elliptical cylinder show the behavior expected in Sec. 1.2.4. For these smooth shapes a higher aspect ratio leads generally to an increased linear response field range.
Even though Fig. 5.9 illustrates the general trend that higher aspect ratios lead to a larger field range of linear response, this trend is not consistently observed across all sampled configurations. Some of these deviations are related to numerical artifacts, such as kinks influencing the determination of the field range of linear response. The irritation for the elliptical cylinder in Fig. 5.9B with $\alpha_{xy} < 8$ is for example related to kinks and further discussed in Fig. 5.10B.
Unrelated to kinks are smaller deviations, for example in horizontal bands of data points. Each of these horizontal bands correspond to shapes with the same y-length and varying x-lengths from $1000 - 2000\,\mathrm{nm}$. According to Sec. 1.6, they are expected to form smooth, monotonic curves with minimal variance and exhibit a slightly positive slope, since increasing x_{len} would continuously increase the aspect ratio. However, they do show noticeable variation. In addition, the continuous optimization of the two shapes show a maximum at $x_{\mathrm{len}} \approx 1930\,\mathrm{nm}$, despite the parameter range extending to $2\,000\,\mathrm{nm}$, a configuration that would correspond to a higher aspect ratio. The underlying physical or numerical mechanisms responsible for this behavior are not yet fully understood.

Furthermore, Sec. 1.2.4 also describes the relationship between the aspect ratio and the slope of the magnetization curve. Fig. 1.4 provides the demagnetization factors for the parameter range defined in Eq. 5.1, with the derivations given in Appendix A. Appendix B presents an analytical simplified expression for the magnetization curve based on a given demagnetization factor. This derivation assumes homogeneous magnetization, a uniform demagnetization field, and magnetization confined to the xy-plane. From these analytical magnetization curves, the label for the analytical field range of linear response, $R_{\mathrm{lin}}^{\mathrm{ana}}$, can be calculated as described in Sec. 5.5. Fig. 5.10 compares $R_{\mathrm{lin}}^{\mathrm{ana}}$ with its numerically computed counterpart, $R_{\mathrm{lin}}^{\mathrm{num}}$.
Fig. 5.10 also uses the dataset defined in Eq. 5.1, consisting of 200 uniformly spaced points in the xy-plane. (A) shows the comparison between $R_{\mathrm{lin}}^{\mathrm{ana}}$ and $R_{\mathrm{lin}}^{\mathrm{num}}$ for a rectangular prism. In red, a linear regression is fitted to the data. Both values are nearly perfectly linearly correlated, with a Pearson correlation coefficient of $r = 0.999$. The regression line has an intercept of $-0.028\,\mathrm{T}$ and a slope of $0.92\,\mathrm{T}$, indicating that the analytical values are systematically too high.
(B) presents the comparison for an elliptical cylinder. Some $R_{\mathrm{lin}}^{\mathrm{num}}$ for the elliptical cylinder are significantly influenced by a kink. The kink-influenced data points are colored in gray, while the unaffected ones are colored in black. For two data points with similar sizes, visualizations of $R_{\mathrm{lin}}^{\mathrm{num}}$ are also presented. These are based on Fig. 5.5 and depict a linear regression of the magnetization curve in yellow, as well as a surrounding margin. The black dashed lines indicate the last continuous point within the margin where $R_{\mathrm{lin}}^{\mathrm{num}}$ is determined. For further details on this process, see Sec. 5.3.1. The main difference between these two magnetization curves is that the upper one shows a kink which remains within the margin around the linear regression, and therefore leads to a larger $R_{\mathrm{lin}}^{\mathrm{num}}$, while the lower magnetization curve exits the margin around the linear regression before the kink reenters the margin. These two magnetization curves showcase how a kink can significantly influence the calculation of $R_{\mathrm{lin}}^{\mathrm{num}}$. For the given dataset, this difference is $\approx 0.018\,\mathrm{T}$. If only the non-kink-influenced data points are fitted, again a near-perfect correlation of $r = 0.999$ and a similar intercept of $-0.028\,\mathrm{T}$ are observed. The slope is however $1.45\,\mathrm{T}$ indication that analytical values are systematically too low.



(A) Rectangular Prism

(B) Elliptical Cylinder

FIGURE 5.10: Comparison between the analytically derived field range of linear response, $R_{\text{lin}}^{\text{ana}}$, and its numerically computed counterpart, $R_{\text{lin}}^{\text{num}}$, for a rectangular prism (A) and an elliptical cylinder (B). Each point corresponds to one of 200 combinations of $[x_{\text{len}}, y_{\text{len}}]$ sampled uniformly in $[1000\text{–}2000,\ 100\text{–}200]$ (nm), with constant $z_{\text{len}} = 20\,\text{nm}$. A red line shows the linear regression fit. (B) shows in black data points unaffected by kinks in the magnetization curve, while those influenced by kinks are shown in gray. Only the data points not influenced by the kink are considered for the regression. Two sections of the magnetization curves for one data point influenced by a kink and one not are also depicted. These sketch the calculation of $R_{\text{lin}}^{\text{num}}$ described in Sec. 5.3.1.

The strong correlation between $R_{\text{lin}}^{\text{ana}}$ and $R_{\text{lin}}^{\text{num}}$ for the rectangular prism and the elliptical cylinder leads to two findings:

1. **Trusting the numerical simulations**
   If the numerical results are assumed to be correct, the theoretical assumptions made in Appendix B are justifiable. That is, if both the energy minimization scheme (state-of-the-art conjugate gradient descent [10], [17]) and the meshing (further analyzed in Sec. 5.5) are assumed to be accurate, then Fig. 5.10 indicates that the rotation of the magnetization occurs only in the xy-plane, and both the demagnetization field and the magnetization can be reasonably approximated as homogeneous in large fraction of the volume.

2. **Trusting the analytical assumptions**
   If the analytical assumptions are assumed to be correct, the numerical approximations are justifiable. This supports the conclusion that in the given parameter range the chosen maximum mesh side length of 5 nm is sufficient to resolve the relevant physical phenomena.

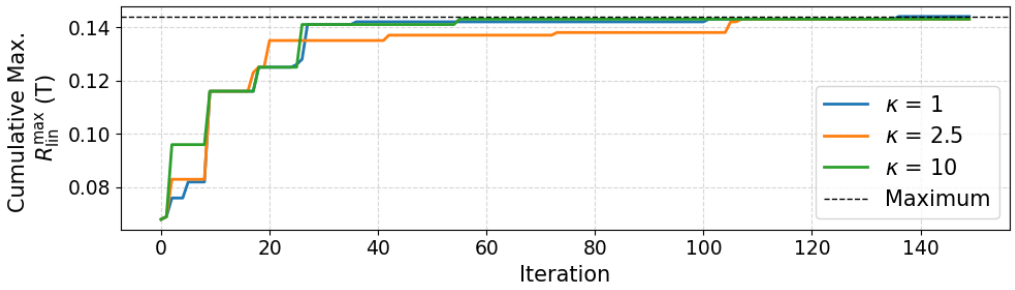
5.4 Hyperparameter Tuning

The suggestion of the next evaluated data point is crucial for the efficiency of the optimization. The Bayesian optimizer uses acquisition functions to suggest these next points based on the initial beliefs about the problem (prior distribution) and the known data points (likelihood function). In Sec. 3.3, three different acquisition functions are introduced: Upper Confidence Bound (UCB), Probability of Improvement (PI), and Expected Improvement (EI). Each of them includes an exploration/-exploitation hyperparameter, denoted by either κ or ξ. An increase in κ or ξ enhances the tendency toward exploitation. As a result, the next evaluation point is typically chosen close to the best-known optimum, thereby focusing on regions where the objective function is already known to be high. Lower values of κ or ξ, in contrast, favor exploration, encouraging sampling in regions where the underlying function is still uncertain and the potential for improvement is high.

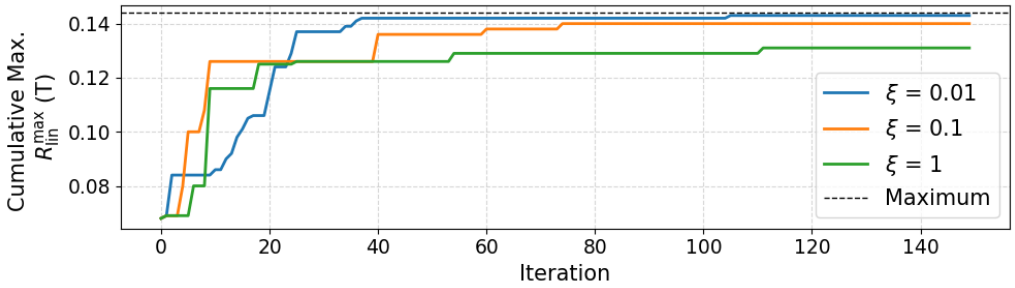
Acquisition Function	**Hyperparameter**	**Value**	R^*_{lin} (T)	**Iterations until** R^*_{lin}
UCB	κ	1	0.144	136
		2.5	0.144	181
		10	0.144	331
PI	ξ	0.01	0.144	181
		0.1	0.144	927
		1	0.137	153
EI	ξ	0	0.144	162
		0.1	0.143	560
		1	0.137	153

TABLE 5.3: Performance of different acquisition functions and exploration/exploitation hyperparameter configurations in Bayesian optimization over 1 000 iterations. The table lists the maximum achieved value R^*_{lin} and the number of iterations required to reach it. κ given in Eq. 3.17 and ξ in 3.19 for PI or Eq. 3.21 for EI.

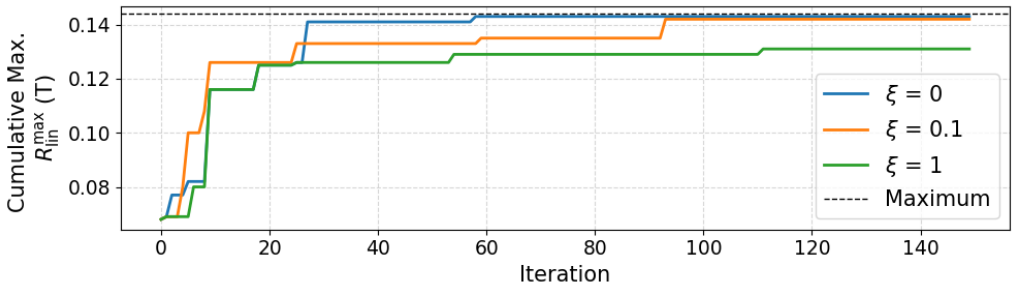
This section provides an overview of how the hyperparameter configuration of the UCB, EI, and PI acquisition functions influences the size optimization of the described thin-film soft magnets. A rectangular prism is used as a example shape throughout this section. Each acquisition function was used to perform optimization over 1 000 iterations in the parameter space described in Tab. 5.1, implying a three-dimensional parameter space, where $z_{\text{len}} \in [10, 20]\,(\text{nm})$ is also defined as a continuous variable. The following optimizations are based on magnetization curves calculated at discrete external field steps. In this section, these field steps are 0.001 T, meaning that the search objective R_{lin} is also discretized to the third decimal place. Tab. 5.3 describes the performance of the acquisition functions with different hyperparameter configurations. The maximum achieved value, R^*_{lin}, was found to be 0.144 T and is in the following assumed to be the global Maximum in the given parameter space.



(A) UCB acquisition function with varying κ configurations.



(B) PI acquisition function with varying ξ configurations.



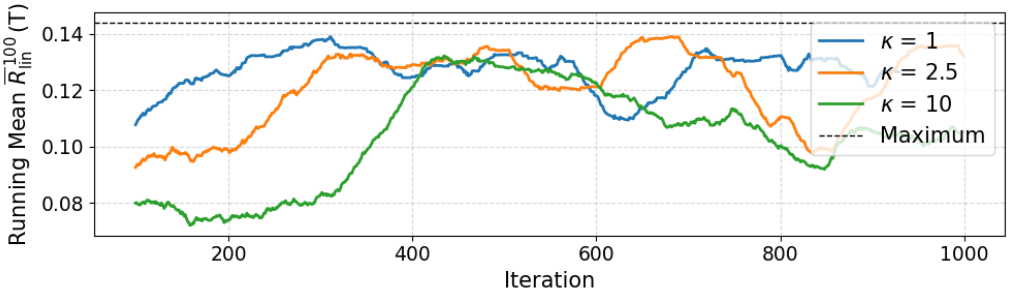
(C) EI acquisition function with varying ξ configurations.

FIGURE 5.11: Comparison of the cumulative maximum $R_{\text{lin}}^{\max}$ for different acquisition functions: Upper Confidence Bound (UCB), Probability of Improvement (PI), and Expected Improvement (EI) across 150 iterations of size optimization of a rectangular prism.

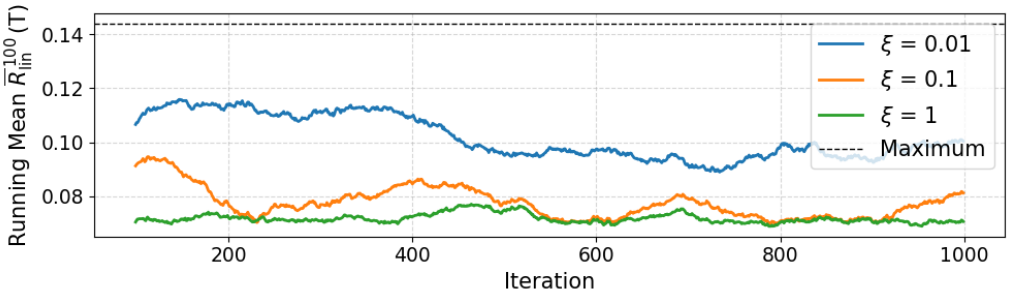
The UCB acquisition function identifies the global maximum for each hyperparameter configuration. For $\kappa = 1$, it identifies the maximum after 136 iterations, which is the overall best performance among all configurations. For $\kappa = 10$, the optimization takes 332 iterations to approximate the global maximum. As shown in Fig. 1.4 the underlying problem is assumed to be rather smoothed and unimodal, exploration is therefore penalized. There should not be other maxima to discover and no need to escape a local one. As a result, $\kappa = 1$ showcases the best performance.
Fig. 5.11A shows, however, that during the first 50 iterations, all three configurations perform similarly. Only after values close to the global maximum are already approximated do the configurations begin to differ, with exploration behavior governed by the choice of κ.

The PI acquisition function configured with $\xi = 0.01$ reaches the global maximum at a comparable rate to the UCB acquisition functions. If the acquisition function is forced to enhance exploration with $\xi = 1$, the global maximum is never reached

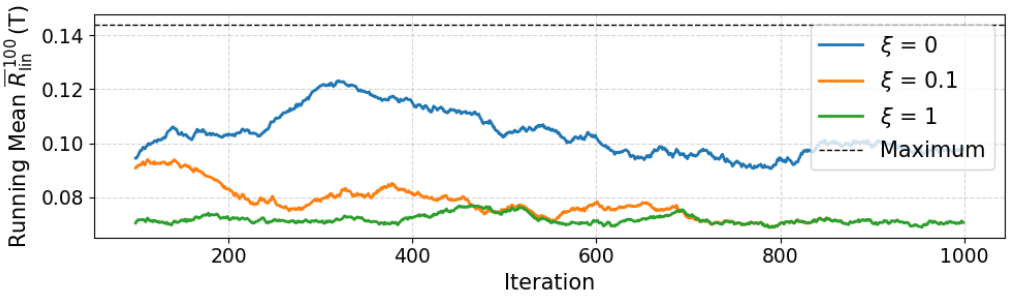
within the 1 000 iterations performed. Fig. 5.11B indicates that although a comparably high label of $R_{\text{lin}} = 0.125\,\text{T}$ is reached after just 29 iterations, the acquisition function does not promote further exploitation over the remaining 971 iterations. Tab. 5.3 and Fig. 5.11B show that the EI acquisition function exhibits similar behavior to PI. If $\xi = 0$, exploitation is enforced and comparable results are achieved. However, for $\xi \geq 0.1$, the global maximum is not reached at all.



(A) UCB acquisition function with varying κ configurations.



(B) PI acquisition function with varying ξ configurations.



(C) EI acquisition function with varying ξ configurations.

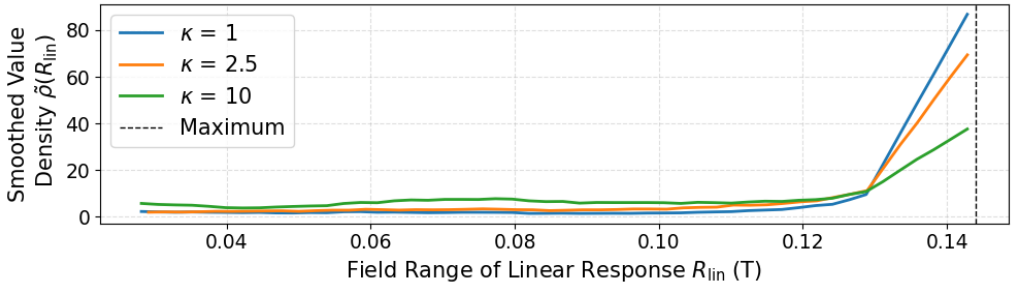
FIGURE 5.12: Comparison of the 100-iteration running mean $\overline{R}^{100}_{\text{lin}}$ of the field range of linear-response across 1 000 Bayesian optimization iterations for the size optimization of a rectangular prism. The acquisition functions used are: Upper Confidence Bound (UCB), Probability of Improvement (PI), and Expected Improvement (EI).

Figure 5.12 provides additional insight into the optimization progress over all 1 000 iterations. It depicts a running mean $\overline{R}^{100}_{\text{lin}}$, calculated with a sliding window of 100 data points. Each point in the plot represents the average of its own value and the 99 surrounding points. All three acquisition functions exhibit the shared characteristic that a lower exploration/exploitation hyperparameter leads to a higher running mean, indicating a stronger focus on exploiting promising regions of the parameter

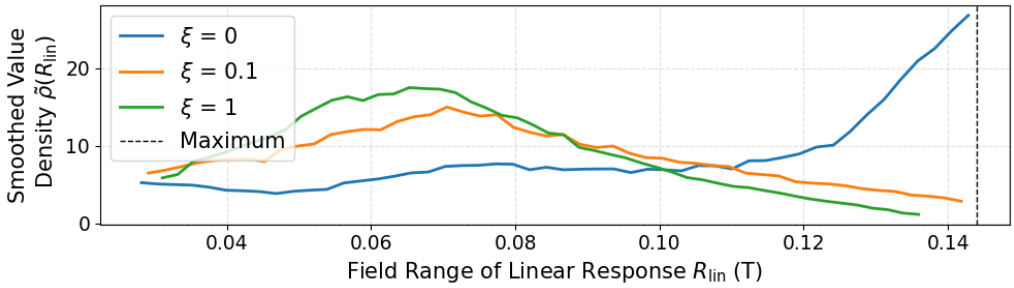
space. This behavior aligns with the theoretical foundation discussed in Sec. 3.3.
Figs. 5.12B and 5.12C especially highlight that the two improvement-based acquisition functions yield a relatively constant $\overline{R}_{\text{lin}}^{100}$, suggesting the absence of clearly defined exploration or exploitation phases. Hyperparameter tuning of ξ instead results in a global shift of the running mean, rather than introducing distinct phases of exploitation or exploration.
Fig. 5.12A indicates that the optimistic optimization policy of the UCB acquisition function exhibits distinct exploration and exploitation phases. While each hyperparameter configuration explores during the first 400 iterations, they all exhibit similar exploitation behavior between iterations 400 and 600, with running means of $\overline{R}_{\text{lin}}^{100} > 0.12\,\text{T}$ for each configuration. Especially in the initial exploration phases of the first 400 Iterations, the UCB acquisition functions also follow the expected behavior that a lower exploration/exploitation hyperparameter leads to a higher running mean.

After Fig. 5.12 revealed initial differences between improvement-driven (PI, EI) and optimistic (UCB) acquisition functions, Fig. 5.13 builds on this analysis. This is achieved by analyzing the statistical distributions of the obtained objective values.
The smoothed value density $\tilde{\rho}\left(R_{\text{lin}}\right)$, representing the field range of linear response across 1 000 iterations, is estimated through a three-step procedure. First, the 1 000 values are binned into 50 equally spaced histogram bins. In the second step, this probability density is smoothed using a running average with a window size of 11. Finally, the resulting estimated probability density function is normalized such that its integral equals 1. Thus, the smoothed density $\tilde{\rho}\left(R_{\text{lin}}\right)$ indicates where objective values are observed more frequently and which are rare.
The difference between improvement-driven and optimistic acquisition functions becomes evident in Fig. 5.13. The UCB acquisition function shows a high-density region at objective values above 0.12 T, while all smaller objective values exhibit equally low likelihoods of being observed. Even when the acquisition function is forced to explore further, this characteristic density profile remains largely unchanged: the likelihood of observing high objective values decreases slightly, while the likelihood of observing values with $R_{\text{lin}} < 0.12\,\text{T}$ increases uniformly.
In contrast, the improvement-driven acquisition functions shown in Figs. 5.13B and 5.13C exhibit density maxima at $R_{\text{lin}} \approx 0.065\,\text{T}$ when configured for exploration. These exploration-oriented versions of EI and PI show no increase in density near the global maximum at $R_{\text{lin}} = 0.144\,\text{T}$, further indicating that they fail to exploit once promising regions are discovered.

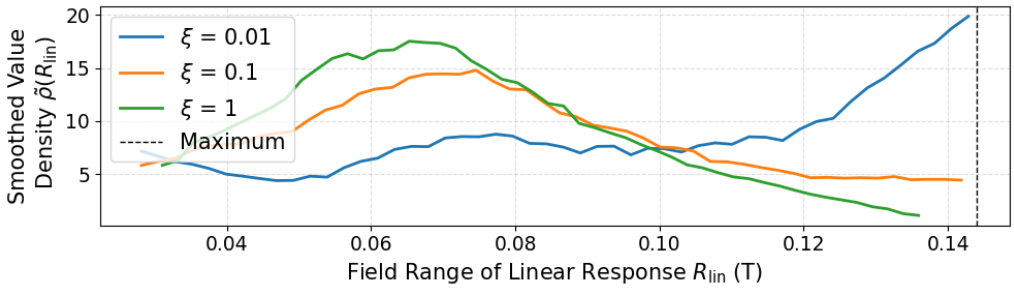
In summary, this section demonstrates the importance of hyperparameter tuning. Tab. 5.3 shows that even minor changes to hyperparameters can significantly affect the number of iterations required to approximate the global maximum.
Figs. 5.11–5.13 extend the theoretical discussion of acquisition functions from Sec. 3.3, providing intuition into how these functions behave under different hyperparameter configurations. Among the acquisition strategies tested in this study, UCB proved to be the most advantageous: it not only delivered the best overall performance, but also appeared robust to hyperparameter tuning. Nonetheless, acquisition functions should always be selected with consideration of the specific characteristics of the target optimization problem.



(A) UCB acquisition function with varying κ configurations.



(B) PI acquisition function with varying ξ configurations.



(C) EI acquisition function with varying ξ configurations.

FIGURE 5.13: Comparison of running means of estimated density functions $\tilde{\rho}(R_{\text{lin}})$ of the sampled field range of linear-response values across 1 000 Bayesian-optimization iterations , smoothed using an 11-point running mean and evaluated using 50 histogram bins. The curves compare the Upper Confidence Bound (UCB), Probability of Improvement (PI), and Expected Improvement (EI) acquisition functions used for optimizing the size of a rectangular prism.

5.5 Meshing Analysis

Finite element energy minimization is performed on discretized objects. However, the free layer of TMR sensors represents a continuous shape. Therefore, the geometry must be discretized with a finite element mesh in order to perform energy minimization and to compute the magnetization curve. The complexity of this energy minimization depends on the number of finite elements used to approximate the continuous geometry. If the mesh is chosen too fine, the computation time becomes unfeasible. Otherwise, if the mesh is too coarse, the physical behavior of the sample is not accurately captured. ℓ_{mesh} describes the maximum side length of a finite element. Although the upper bound is not linearly dependent on the number of finite elements, it still affects their total count. Each finite element is assumed to

have a uniform magnetization. If the finite element size is too large relative to the characteristic length scale of magnetization variation in the material, the simulation loses its accuracy.
Fig. 5.14 visualizes the influence of mesh resolution on the magnetization curve. A rectangular prism with dimensions $x_{\text{len}} = 1000\,\text{nm}$, $y_{\text{len}} = 100\,\text{nm}$, and $z_{\text{len}} = 10\,\text{nm}$ serves as an example, while the maximum finite element length was iteratively reduced in the interval $\ell_{\text{mesh}} \in [0.9, 30]$ (nm).

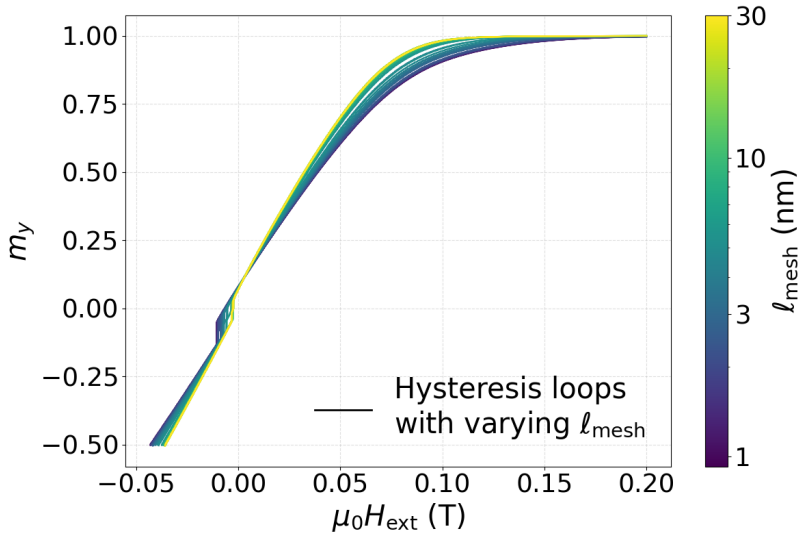


FIGURE 5.14: Magnetization curves with varying maximum finite element side length ℓ_{mesh}. 82 simulations of a rectangular prism with size $x_{\text{len}} = 1000\,\text{nm}$, $y_{\text{len}} = 100\,\text{nm}$, and $z_{\text{len}} = 10\,\text{nm}$, while varying ℓ_{mesh} across the interval $[0.9, 30]$ (nm) are displayed.

Fig. 5.14 provides two key insights. First, the kink observed at $H_{\text{ext}} \approx -0.01\,\text{T}$ shifts in position depending on the mesh resolution. As identified in Sec. 5.2, this kink is associated with the discrete switching of the magnetization at the end faces of the prism. Fig. 5.14 indicates that this effect is mesh-dependent: coarser meshes shift the kink to lower external field amplitudes.
Second, reducing the maximum mesh size alters the behavior of the magnetization curve, leading to a reduced slope in the hysteresis loop. The reduced slope of the magnetization curve affects the optimization objective and is therefore critical to the optimization in this study.

Fig. 5.15 demonstrates that ℓ_{mesh} has a significant influence on R_{lin}. Varying ℓ_{mesh} within the interval $[0.78, 30]\,\text{nm}$ changes R_{lin} by up to $0.012\,\text{T}$, introducing a unfeasible error margin. Fig. 5.15 also illustrates that for $\ell_{\text{mesh}} > 10\,\text{nm}$, no consistent trend in R_{lin} can be observed. Although the variance remains high in this region, no general pattern can be identified. This behavior is likely linked to the geometry of the example shape, a rectangular prism with its smallest dimension in the z-direction, $z_{\text{len}} = 10\,\text{nm}$. Thus, the finite element size in the z-direction cannot exceed $10\,\text{nm}$. This suggests that the number of finite elements along the direction of the smallest geometric dimension may be critical for accurately approximating the magnetization curve.

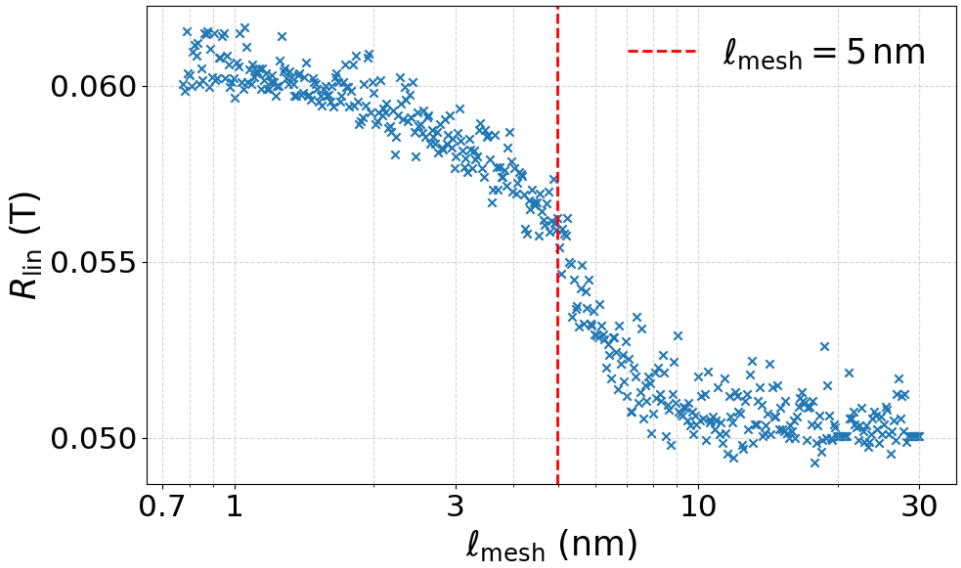


FIGURE 5.15: Influence of the maximum finite element side length ℓ_{mesh} on the field range of linear response R_{lin}. A total of 457 simulations of a rectangular prism with dimensions $x_{\text{len}} = 1000\,\text{nm}$, $y_{\text{len}} = 100\,\text{nm}$, and $z_{\text{len}} = 10\,\text{nm}$ are shown, as ℓ_{mesh} is logarithmically varied over the interval across the interval $[0.78, 30]$ (nm). A red dashed line indicates the maximum side length equal to the exchange length, as defined in Eq. 5.2.

For $\ell_{\text{mesh}} < 2\,\text{nm}$, at least 10 finite elements are present in each spatial dimension. In this parameter region, the resulting R_{lin} stabilizes, indicating numerical convergence. As previously described, finite element simulations are expected to converge toward the true solution as the mesh size tends towards zero. Therefore, the true field range of linear response for the example shape is estimated to converge toward $\approx 0.061\,\text{T}$. However, Fig. 5.15 does not provide a sufficient number of data points to support a detailed convergence analysis.

The meshes resulting from $\ell_{\text{mesh}} < 2\,\text{nm}$, exceed 10 million finite elements, and Fig. 5.16 demonstrates how the resulting simulation time increases significantly—from 5 min for $\ell_{\text{mesh}} = 5\,\text{nm}$ to 7.45 h for $\ell_{\text{mesh}} = 0.9\,\text{nm}$. These simulations were performed on a high-performance computing (HPC) cluster using a single node with 4 dedicated cores (Intel Xeon E5-2680 v4, 2.40 GHz, 28 physical cores total) and 50 GB of RAM. Also an NVIDIA Tesla P100-PCIE-16GB GPU with 16 GB of VRAM was used.

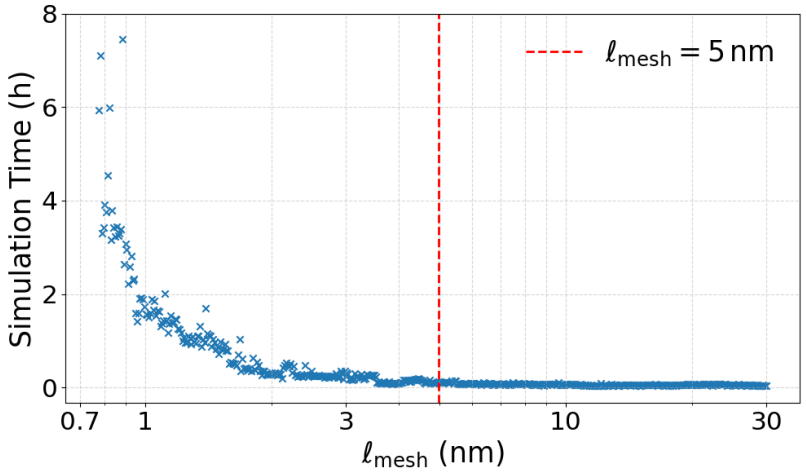


FIGURE 5.16: Influence of the maximum finite element side length ℓ_{mesh} on the simulation time. A total of 457 simulations of a rectangular prism with dimensions $x_{\text{len}} = 1000\,\text{nm}$, $y_{\text{len}} = 100\,\text{nm}$, and $z_{\text{len}} = 10\,\text{nm}$ are shown, as ℓ_{mesh} is logarithmically varied over the interval across the interval $[0.78, 30]$ (nm). A red dashed line indicates the maximum side length equal to the exchange length, as defined in Eq. 5.2. A single node with 4 dedicated cores (Intel Xeon E5-2680 v4, 2.40 GHz, 28 physical cores total) and 50 GB of RAM and an NVIDIA Tesla P100-PCIE-16GB GPU with 16 GB of VRAM as used.

Figs. 5.15 and 5.16 indicate that mesh sizes close to the assumed convergence value of R_{lin} result in unfeasible simulation times for performing several thousand iterations, as performed for example in Sec. 5.4. A commonly used upper bound for the finite element size is the exchange length ℓ_{ex}, which characterizes the length scale over which the magnetization varies smoothly. For $Ni_{80}Fe_{20}$, the exchange length can be calculated as follows [8]:

$$\ell_{\text{ex}} = \sqrt{\frac{2A}{\mu_0 M_s^2}} = \sqrt{\frac{2 \times (1.0 \times 10^{-11}\,\text{J/m})}{(4\pi \times 10^{-7}\,\text{H/m})\,(8.0 \times 10^5\,\text{A/m})^2}} \approx 5\,\text{nm}. \tag{5.2}$$

The finite element side length $\ell_{\text{mesh}} = 5\,\text{nm}$, indicated by the red dashed line in Fig. 5.15, does not correspond to full convergence of R_{lin}. Nonetheless, this work uses $\ell_{\text{mesh}} = 5\,\text{nm}$ for all simulations presented, under the assumption that the resulting R_{lin} values differ from the fully converged values by a consistent linear offset. This hypothesis is supported by Fig. 5.10, which shows that even for $\ell_{\text{mesh}} = 5\,\text{nm}$, the numerically approximated values $R_{\text{lin}}^{\text{num}}$ are linearly correlated with the analytically derived values $R_{\text{lin}}^{\text{ana}}$.

This hypothesis remains to be examined further. Extruded meshes in the z-dimension or reduced-order models [40] could offer promising approaches for the computation of $R_{\text{lin}}^{\text{num}}$ within computationally feasible runtime.

Chapter 6

Summary and Outlook

The presented study introduces an open-source simulation framework for optimizing the hysteresis curves of magnetic samples. A focus is directed toward $Ni_{80}Fe_{20}$ thin-film soft magnets, intended to act as the free layers in tunneling magnetoresistance (TMR) sensors.
The software framework has a low entry threshold to cover a wide range of use cases in research and industry. The modular, object-oriented design enables features of the hysteresis curve as a search objective, as well as geometric configuration and manufacturing capabilities. To outline the capabilities of the software framework, three shapes are continuously optimized in their dimensions to achieve the maximal field range of linear response to changes in the external field. An elliptical cylinder with a major axis of 1927 nm, a minor axis of 100 nm, and an extrusion height of 20 nm achieved a field range of linear response of 0.172 T by reducing the external field from 1 T to 0 T. The optimization is performed in the parameter range [1000–2000, 100–200, 20] (nm) and each new size is suggested by a Bayesian optimizer. Possible search objectives in further optimizations could be, for example, to include interval conditions, the manufacturing cost or an increased smoothness of the hysteresis curves, penalizing any kinks.
This study contributes to the European research project MaMMoS [1], which aims to develop an integrated simulation suite for modeling magnetic devices from the atomic scale up to the device scale. The presented software framework is integrated into the MaMMoS suite at the device scale.

Besides these contributions to the MaMMoS suite, several general scientific findings about the numerical optimization of thin-film soft magnets are presented. In Sec. 1.2.4 analytical expressions for the demagnetization factors of elliptical cylinders and rectangular prisms are developed. These expressions allow for an analytical approximation of the hysteresis curves for the two considered shapes, treating their aspect ratio as a variable. The results indicate that an increased aspect ratio corresponds to a reduced slope in the hysteresis loop. Section 5.3.5 benchmarks these approximations against finite-element micromagnetic simulations for both geometries, confirming that the field range of linear response grows with increasing aspect ratio. Furthermore, a near-perfect linear correlation between the analytical approximation of the field range of linear response and the numerical results is observed. However, a rectangular prism with tapered ends shows a different optimum, with a smaller aspect ratio, demonstrating that the trend established for untapered prisms and cylinders does not generalize to all shapes.
The study also provides an overview of Bayesian optimization and its applicability for micromagnetic optimizations, presenting convergence after 136 iterations of size

optimization, though exhibiting high sensitivity to hyperparameter tuning.
The maximum side length of finite element meshes for soft-magnetic thin-film elements was also analyzed. The characteristic exchange length of $Ni_{80}Fe_{20}$ at 5 nm as a maximum finite element side length led to a numerically near-perfect correlation between the numerical and analytical approximations of the field range of linear response. However, further reducing the maximum finite element side length still influences the magnetization curves of the samples. Therefore, further meshing strategies remain to be explored.

Future work may focus on developing kink-sensitive optimization objectives and comparing the analytical and numerical findings to experimental measurements of fabricated samples to further validate the findings. Additionally, a systematic analysis of how varying mesh resolutions affect the correlation between analytical estimates and numerical results for the field range of linear response would provide further insights into soft-magnetic thin-film simulations.

Appendix A

Analytic Expressions of Demagnetizing Factors

An analytic expression of the demagnetization factor for a rectangular prism is discussed in detail in [13]. The formula used to calculate the values shown in Fig. 1.3 and 1.4A is given below as Eq. A.1. x, y, and z are the side lengths of the rectangular prism, and the demagnetization factor D_z is always calculated along the z-axis. However A.1 does not apply any restriction on in the aspect ratios of x, y and z therefore D_x and D_y can be calculated tho rotations of the prism or via Eq. 1.9.

$$\begin{aligned}\pi D_z = &\frac{b^2-c^2}{2bc}\ln\left(\frac{\sqrt{a^2+b^2+c^2}-a}{\sqrt{a^2+b^2+c^2}+a}\right)+\frac{a^2-c^2}{2ac}\ln\left(\frac{\sqrt{a^2+b^2+c^2}-b}{\sqrt{a^2+b^2+c^2}+b}\right)\\&+\frac{b}{2c}\ln\left(\frac{\sqrt{a^2+b^2}+a}{\sqrt{a^2+b^2}-a}\right)+\frac{c}{2a}\ln\left(\frac{\sqrt{b^2+c^2}-b}{\sqrt{b^2+c^2}+b}\right)\\&+\frac{c}{2b}\ln\left(\frac{\sqrt{a^2+c^2}-a}{\sqrt{a^2+c^2}+a}\right)+2\arctan\left(\frac{ab}{c\sqrt{a^2+b^2+c^2}}\right)\\&+\frac{a^3+b^3-2c^3}{3abc}+\frac{a^2+b^2-2c^2}{3abc}\sqrt{a^2+b^2+c^2}\\&+\frac{c}{ab}(\sqrt{a^2+c^2}+\sqrt{b^2+c^2})-\frac{(a^2+b^2)^{3/2}+(c^2+b^2)^{3/2}+(a^2+c^2)^{3/2}}{3abc}\end{aligned} \tag{A.1}$$

Fig. 1.4B presents an analytic calculation of the demagnetization factor N_y for an elliptical cylinder. The factor N_y is computed using the relation: $N_y = 1 - N_x - N_z$ Analytic expressions for N_x and N_z are derived from the work of [14].

In this model:

a, b - Major and minor semi-axes of the elliptical cross-section, with $a \geq b$.

z - Cylinder height.

ϕ - Angular coordinate in the elliptical cross-section.

${}_2F_1$ - Gauss hypergeometric function.

The demagnetization factor (N_z) is given by:

$$N_z = 1 + \frac{8}{3\pi^2\xi}K(e^2) - \frac{2}{\pi}\int_0^{\pi/2}\sqrt{1-e^2\cos^2\phi}\,{}_2F_1\left(-\frac{1}{2},\frac{1}{2},2,-\frac{1}{\xi^2 g^2}\right)d\phi. \tag{A.2}$$

Similarly, the demagnetization factor (N_x) is:

$$N_x = \frac{8}{3\pi^2\xi e^2}\left(\beta^2 K(e^2) - E(e^2)\right) + \frac{2\beta^2}{\pi}\int_0^{\pi/2} \frac{\cos^2\phi}{g^2}\, {}_2F_1\left(-\frac{1}{2}, \frac{1}{2}, 2, -\frac{1}{\xi^2 g^2}\right) d\phi. \tag{A.3}$$

where:

$$K(e^2) = \int_0^{\pi/2} \frac{d\theta}{\sqrt{1 - e^2\sin^2\theta}}, \quad E(e^2) = \int_0^{\pi/2} \sqrt{1 - e^2\sin^2\theta}\, d\theta. \tag{A.4}$$

The parameters used in these equations are:

$$\xi = \frac{t}{2b}, \quad e = \sqrt{1 - \frac{b^2}{a^2}}, \quad \beta = \frac{b}{a}, \quad g = \sqrt{1 - e^2\cos^2\phi}. \tag{A.5}$$

Appendix B

Analytic Formulation of the Hysteresis Loop

The aim of this section is to develop an analytical approximation of a hysteresis loop for soft magnets. To achieve this, three major assumptions are made:

1. Homogeneous magnetization throughout the entire sample.
2. Homogeneous demagnetization fields.
3. The magnetization rotates only in the xy-plane and has no z-component.

According to Sec. 1.2, the total free energy of a magnetic sample is given by:

$$\mathcal{E}_{\text{tot}} = \int_V \Big((\nabla \mathbf{m})^2 + \mathcal{E}_a - \mu_0 M_s \, (\mathbf{m} \cdot \mathbf{H}_{\text{ext}}) - \tfrac{1}{2} \mu_0 M_s \, (\mathbf{m} \cdot \mathbf{H}_{\text{demag}}) \Big) \, dV. \tag{B.1}$$

For soft magnets, the anisotropy energy $\mathcal{E}_a$, which includes magnetocrystalline and induced anisotropy, can be neglected [12], while shape anisotropy is accounted for in the demagnetization terms. Assumption 1 implies that if the magnetization is homogeneous, then $\nabla \mathbf{m} = 0$. Substituting this into Eq. (B.1) gives

$$\mathcal{E}_{\text{tot}} = \int_V \Big(- \mu_0 M_s \, (\mathbf{m} \cdot \mathbf{H}_{\text{ext}}) - \tfrac{1}{2} \mu_0 M_s \, (\mathbf{m} \cdot \mathbf{H}_{\text{demag}}) \Big) \, dV. \tag{B.2}$$

A uniform magnetization leads to a uniform energy density, reducing the total energy to

$$\mathcal{E}_{\text{tot}} = V \Big(- \mu_0 M_s \, (\mathbf{m} \cdot \mathbf{H}_{\text{ext}}) - \tfrac{1}{2} \mu_0 M_s \, (\mathbf{m} \cdot \mathbf{H}_{\text{demag}}) \Big), \tag{B.3}$$

$$\frac{\mathcal{E}_{\text{tot}}}{V \mu_0} = - M_s \, (\mathbf{m} \cdot \mathbf{H}_{\text{ext}}) - \tfrac{1}{2} M_s \big(H_x M_x + H_y M_y \big). \tag{B.4}$$

Equation 1.7 holds under Assumption 2 and allows the fields to be written as

$$H_i = - N_i M_i, \quad i \in \{x, y, z\}. \tag{B.5}$$

$$\frac{\mathcal{E}_{\text{tot}}}{V \mu_0} = - M_s \, (\mathbf{m} \cdot \mathbf{H}_{\text{ext}}) - \tfrac{1}{2} \Big(M_x \, (-N_x M_x) + M_y \, (-N_y M_y) \Big) \tag{B.6}$$

Assumption 3 enforces planar magnetization in xy. Let ϕ denote the angle between $\mathbf{m}$ and the external field $\mathbf{H}_{\text{ext}}$, assumed along y. Then

$$m_y = \cos\phi, \quad M_x = M_s \, \sin\phi, \quad M_y = M_s \, \cos\phi, \quad \mathbf{m} \cdot \mathbf{H}_{\text{ext}} = |\mathbf{H}_{\text{ext}}| \, \cos\phi.$$

Substitution yields

$$\frac{\mathcal{E}_{\text{tot}}}{V\,\mu_0} = -\,M_s\,|\mathbf{H}_{\text{ext}}|\,\cos\phi - \tfrac{1}{2}\,M_s^2\big(\sin^2\phi\,N_x + \cos^2\phi\,N_y\big). \tag{B.7}$$

Using the identity $\cos^2\phi = 1 - \sin^2\phi$ evolves

$$\frac{\mathcal{E}_{\text{tot}}}{V\,\mu_0} = -\,M_s\,|\mathbf{H}_{\text{ext}}|\,\cos\phi - \tfrac{1}{2}\,M_s^2\,(\sin^2\phi\,(N_x - N_y) + N_y). \tag{B.8}$$

The equilibrium state follows by setting

$$\tfrac{d\mathcal{E}_{\text{tot}}}{d\phi} = 0.$$

This condition leads to

$$-\frac{1}{V\,\mu_0}\frac{d\mathcal{E}_{\text{tot}}}{d\phi} = M_s\,|\mathbf{H}_{\text{ext}}|\,\sin\phi + M_s^2\,\sin\phi\,\cos\phi\,(N_x - N_y) \stackrel{!}{=} 0. \tag{B.9}$$

For $\sin\phi \neq 0$, it follows that

$$\cos\phi = \frac{|\mathbf{H}_{\text{ext}}|}{M_s\,(N_y - N_x)}, \tag{B.10}$$

giving

$$\phi(\mathbf{H}_{\text{ext}}) = \cos^{-1}\!\Big(\tfrac{|\mathbf{H}_{\text{ext}}|}{M_s\,(N_y - N_x)}\Big). \tag{B.11}$$

Since $|\cos\phi| \leq 1$, the magnitude of the field must satisfy

$$\big|\mathbf{H}_{\text{ext}}\big| \leq M_s\,\big|N_y - N_x\big|.$$

Hence the solution is piecewise:

$$\phi(H_{\text{ext}}) = \begin{cases} \cos^{-1}\!\Big(-\tfrac{H_{\text{ext}}}{M_s\,(N_y - N_x)}\Big), & \text{if } \big|\tfrac{H_{\text{ext}}}{M_s\,(N_y - N_x)}\big| \leq 1, \\ 0 \text{ or } \pi, & \text{otherwise (full saturation).} \end{cases} \tag{B.12}$$

The dimensionless y-component of the magnetization is

$$m_y = \cos\phi = \begin{cases} -\,\tfrac{H_{\text{ext}}}{M_s\,(N_y - N_x)}, & \text{if } \big|\tfrac{H_{\text{ext}}}{M_s\,(N_y - N_x)}\big| \leq 1, \\ \pm 1, & \text{otherwise (full saturation).} \end{cases} \tag{B.13}$$

Any violation of $|m_y| \leq 1$ implies full saturation ($|m_y| = 1$) corresponding to magnetization aligned parallel or antiparallel to $\mathbf{H}_{\text{ext}}$.
Under the three major assumptions (homogeneous magnetization, homogeneous demagnetization field, and planar rotation), Eq. (B.11) and the piecewise condition in Eq. (B.13) provide an analytic expression for the hysteresis loop.

The given assumptions and rearrangements simplify the derivation; however, real materials exhibit magnetization variations, nonuniform demagnetizing fields, and out-of-plane components. Therefore, this finding represents an approximation rather than a full representation of actual hysteresis behavior.

Appendix C

Sample Configuration

A configuration file for the simulation framework in Chapter 4 is depicted below.

```
database:
  use_db: False
  read: False
  write: False
  name: "example"
  db_path: '/.../example.db'
  post_proc_global_path: '/.../example'
shape:
  name: "Box"            # Ellipse, Box, Stick
  init_x: 0.3
  init_y: 0.1
  init_z: 0.01
simulation:
  sim_name: 'example'
  iter: 5
  main_mesh_min: 1e-4
  main_mesh_max: 1.0
  airbox_mesh_max: 5e-3
  x_start: 0.3
  x_stop: 0.4
  y_start: 0.1
  y_stop: 0.2
  z_start: 0.01
  z_stop: 0.02
  hstart: 0.5
  hfinal: -0.1
  hstep: -0.001
optimizer:
  acq_kind: 'ei'           # 'ei', 'ucb', 'poi'
  kappa: 0.0
  xi: 25.0
  kappa_decay: 1.0
  kappa_decay_delay: 0
server:
  number_cores: 2
  mem_GB: 25
  gpu: 'nv12'
general_settings:
  log_level: 2
  location: ''
```

LISTING C.1: Sample Configuration

The provided configuration file includes settings grouped into categories, each controlling specific aspects of the optimization process. The database parameters

determine whether the simulation should interact with a database (*use_db*), specify whether reading and writing to the database are allowed (*read*, *write*), set the database's name (*name*), define its absolute path (*db_path*), and establish a global path for saving post-processing plots (*post_proc_global_path*). These are the only global paths that have to be defined since they are independent of the optimization circuits and therefore can be relative to the optimization folder.
The *shape* section configures the initial geometry used in simulations. A rectangular prism, an elliptical cylinder, and an extruded stick can be configured as *name*. Their initial dimensions along the x-, y-, and z-axes are set as *init_x*, *init_y*, *init_z*. An optimization setup for three parameters has been configured. These can, however, be adjusted to the specific shapes as well.
Within *simulation*, the settings control the specific simulation scenario (*sim_name*), the number of optimization iterations (*iter*), and meshing resolution. *main_mesh_min* defines the minimum edge length of mesh cells in the sensor element, and *main_mesh_max* defines the maximum edge length. *airbox_mesh_max* configures the maximum edge length of the airbox simulated around the mesh object. The start and end values for parameter sweeps are defined as *x_start*, *x_stop*, *y_start*, *y_stop*, *z_start*, *z_stop*, and the configuration of the magnetization curve range (*hstart*, *hfinal*) and step size (*hstep*).
The *optimizer* parameters dictate the type of acquisition function *acq_kind* used in optimization (UCB, EI, PI), alongside tuning parameters such as exploration intensity *kappa*, improvement threshold *xi*, decay rate for exploration *kappa_decay*, and delay before decay begins *kappa_decay_delay*.
In the server settings, computational resources for running simulations are configured, including the number of CPU cores *number_cores*, available memory in GB *mem_GB*, and the GPU type *gpu*.
Finally, *general_settings* defines the logging verbosity *log_level* and a placeholder for specifying execution location *location*, which is explicitly set during execution.

Appendix D

Deutschsprachige Zusammenfassung

Es wird ein Open-Source-basiertes Simulationsframework zur Modellierung und Optimierung von Hysteresekurven magnetischer Sensorelemente präsentiert. Dabei liegt der Fokus auf Dünnschicht-Weichmagneten aus $Ni_{80}Fe_{20}$, welche sich in magnetischen Tunnelwiderstandssensoren (TMR-Sensoren) möglichst frei in Richtung eines externen Feldes ausrichten sollen.
Das vorgestellte Simulationsframework bietet eine niedrigschwellige Möglichkeit, für die Optimierung magnetischer Sensoren, sowohl in der Forschung als auch in der Industrie. Es ermöglicht, Merkmale der Hysteresekurve ebenso als Optimierungsziel zu definieren wie geometrische Konfigurationen und Herstellungsparameter. Um die Leistungsfähigkeit des Frameworks zu demonstrieren, werden drei Geometrien hinsichtlich ihrer Dimensionen kontinuierlich optimiert. Ziel der Optimierung ist ein Design, dessen magnetische Momente über einen möglichst breiten Feldbereich linear auf externe Magnetfeldänderungen reagieren. Als bestes Design wurde ein elliptischer Zylinder mit einer Hauptachse von 1927 nm, einer Nebenachse von 100 nm und einer Höhe von 20 nm identifiziert. Dieser zeigte beim Absenken des externen Magnetfeldes von 1 T auf 0 T einen linearen Reaktionsbereich von 0,172 T. Die Optimierung erfolgt im Parameterbereich $[1000–2000,\ 100–200,\ 20–10]$ (nm), wobei die neuen Dimensionen einer Geometrie von einem Bayesschen Optimierer vorgeschlagen werden.

Diese Arbeit trägt zum europäischen Forschungsprojekt MaMMoS [1] bei, das die Entwicklung eines integrierten Software-Framework zur Modellierung magnetischer Bauelemente von der atomaren bis zur Komponentenebene verfolgt. Die vorgestellte Arbeit ist auf der Komponentenebene in das MaMMoS-Framework integriert.

Neben diesen Beiträgen zum MaMMoS-Framework werden mehrere allgemeine wissenschaftliche Erkenntnisse zur numerischen Optimierung von Dünnschicht-Weichmagneten präsentiert. In Abschnitt 1.2.4 werden analytische Formulierungen für die Entmagnetisierungsfaktoren elliptischer Zylinder und rechteckiger Prismen entwickelt. Diese ermöglichen eine analytische Näherung der Hysteresekurven für die beiden betrachteten Geometrien, wobei das Aspektverhältnis zwischen den Längen in x- und y-Richtung als Variable behandelt wird. Die Ergebnisse zeigen, dass ein höheres Aspektverhältnis mit einer geringeren Steigung in der Hystereseschleife einhergeht. Abschnitt 5.3.5 vergleicht diese Vorhersagen mit finite-elemente-basierten mikromagnetischen Simulationen beider Geometrien und

bestätigt, dass der lineare Reaktionsbereich mit zunehmendem Aspektverhältnis ansteigt. Darüber hinaus besteht eine nahezu perfekte lineare Übereinstimmung zwischen der analytischen Abschätzung des linearen Reaktionsbereichs und den Ergebnissen der numerischen Simulation. Ein rechteckiges Prisma mit zulaufenden Enden weist jedoch ein anderes Optimum bei geringerem Aspektverhältnis auf, was verdeutlicht, dass der für rechteckige Prismen und Zylinder etablierte Trend nicht generell auf andere Geometrien übertragbar ist.
Die Studie bietet zudem einen Überblick über die Bayessche Optimierung und deren Anwendbarkeit auf die mikromagnetische Geometrieoptimierung. Es wird eine Konvergenz nach 136 Iterationen der Größenoptimierung erreicht, wobei eine hohe Sensitivität gegenüber der Hyperparameterwahl festgestellt wird.
Darüber hinaus wurde die maximale Kantenlänge der finiten Elemente im Rahmen der Diskretisierung untersucht. Die charakteristische Austauschlänge von $Ni_{80}Fe_{20}$ beträgt 5 nm; diese Kantenlänge ermöglicht eine Bestätigung der analytischen Vorhersagen. Eine weitere Reduktion der maximalen Kantenlänge beeinflusst jedoch weiterhin die Form der Hysteresekurven, weshalb zusätzliche Vermaschungsstrategien weiter untersucht werden sollten.

Bibliography

[1] *Mammos – magnetic materials for miniaturized optical systems*, [Online]. Available: https://cordis.europa.eu/project/id/101135546. Accessed: Feb. 26, 2025, European Commission, 2024.

[2] O. Gutfleisch, M. A. Willard, E. Brück, C. H. Chen, S. Sankar, and J. P. Liu, "Magnetic materials and devices for the 21st century: Stronger, lighter, and more energy efficient," *Adv. Mater.*, vol. 23, no. 7, pp. 821–842, May 2011.

[3] J. Leliaert and J. Mulkers, "Tomorrow's micromagnetic simulations," *J. Appl. Phys.*, vol. 125, no. 18, May 2019.

[4] N. A. Wibowo, C. Kurniawan, D. K. Kusumahastuti, A. Setiawan, and E. Suharyadi, "Potential of tunneling magnetoresistance coupled to iron oxide nanoparticles as a novel transducer for biosensors-on-chip," *J. Electrochem. Soc.*, vol. 171, no. 1, p. 017 512, Jan. 2024.

[5] X. Liu, K. Lam, K. Zhu, *et al.*, "Overview of spintronic sensors with internet of things for smart living," *IEEE Trans. Magn.*, vol. 55, no. 11, pp. 1–22, Aug. 2019.

[6] R. B. Diba, M. L. Plumer, and I. Saika-Voivod, "Simulating micromagnetism," *Adv. Phys. X*, vol. 9, no. 1, p. 2 409 070, Oct. 2024.

[7] S. Amini, I. Vannieuwenhuyse, and A. Morales-Hernández, "Constrained bayesian optimization: A review," *IEEE Access*, vol. 13, pp. 1581–1593, Dec. 2025.

[8] J. M. D. Coey, *Magnetism and Magnetic Materials*. Cambridge, U.K.: Cambridge Univ. Press, 2010.

[9] D. Jiles, *Introduction to magnetism and magnetic materials*. CRC press, 2015.

[10] L. Exl, "Tensor grid methods for micromagnetic simulations," Ph.D. dissertation, Technische Univ. Wien, Vienna, Austria, 2014.

[11] L. Exl, D. Suess, and T. Schrefl, "Micromagnetism," in *Handbook of Magnetism and Magnetic Materials*, M. Coey and S. Parkin, Eds. Cham, Switzerland: Springer, 2020, pp. 1–44.

[12] L. Yin, D. Wei, N. Lei, *et al.*, "Magnetocrystalline anisotropy in permalloy revisited," *Phys. Rev. Lett.*, vol. 97, no. 6, p. 067 203, Aug. 2006.

[13] A. Aharoni, "Demagnetizing factors for rectangular ferromagnetic prisms," *J. Appl. Phys.*, vol. 83, no. 6, pp. 3432–3434, Mar. 1998.

[14] M. Beleggia, M. De Graef, Y. T. Millev, D. A. Goode, and G. Rowlands, "Demagnetization factors for elliptic cylinders," *J. Phys. D: Appl. Phys.*, vol. 38, no. 18, p. 3333, Sep. 2005.

[15] R. Cowburn and M. Welland, "Analytical micromagnetics of near single domain particles," *J. Appl. Phys.*, vol. 86, no. 2, pp. 1035–1040, Jul. 1999.

[16] J. Fidler, T. Schrefl, W. Scholz, D. Suess, R. Dittrich, and M. Kirschner, "Micromagnetic modelling and magnetization processes," *J. Magn. Magn. Mater.*, vol. 272, pp. 641–646, May 2004.

[17] J. Fischbacher, A. Kovacs, H. Oezelt, *et al.*, "Nonlinear conjugate gradient methods in micromagnetics," *Aip Adv.*, vol. 7, no. 4, Apr. 2017.

[18] D. C. Liu and J. Nocedal, "On the limited memory bfgs method for large scale optimization," *Mathematical programming*, vol. 45, no. 1, pp. 503–528, Aug. 1989.

[19] M. R. Hestenes, E. Stiefel, *et al.*, *Methods of conjugate gradients for solving linear systems*. NBS Washington, DC, 1952, vol. 49.

[20] S. Schaffer, T. Schrefl, H. Oezelt, N. J. Mauser, and L. Exl, *Physics aware machine learning for micromagnetic energy minimization: Recent algorithmic developments*, arXiv:2409.12877, 2024.

[21] H. Forster, T. Schrefl, R. Dittrich, W. Scholz, and J. Fidler, "Fast boundary methods for magnetostatic interactions in micromagnetics," *IEEE Trans. Magn.*, vol. 39, no. 5, pp. 2513–2515, Sep. 2003.

[22] L. Exl, J. Fischbacher, A. Kovacs, H. Oezelt, M. Gusenbauer, and T. Schrefl, "Preconditioned nonlinear conjugate gradient method for micromagnetic energy minimization," *Comput. Phys. Commun.*, vol. 235, pp. 179–186, Feb. 2019.

[23] J. R. Shewchuk *et al.*, "An introduction to the conjugate gradient method without the agonizing pain," Carnegie Mellon University, Dept. of Computer Science, Pittsburgh, PA, USA, Tech. Rep., 1994.

[24] S. Yan, Z. Zhou, Y. Yang, Q. Leng, and W. Zhao, "Developments and applications of tunneling magnetoresistance sensors," *Tsinghua Sci. Technol.*, vol. 27, no. 3, pp. 443–454, Jun. 2021.

[25] M. Julliere, "Tunneling between ferromagnetic films," *Phys. Lett. A*, vol. 54, no. 3, pp. 225–226, Sep. 1975.

[26] T. Miyazaki and N. Tezuka, "Giant magnetic tunneling effect in fe/al2o3/fe junction," *J. Magn. Magn. Mater.*, vol. 139, no. 3, pp. L231–L234, Jan. 1995.

[27] L. A. Francis and K. Poletkin, *Magnetic sensors and devices: technologies and applications*. CRC Press, 2017.

[28] D. R. Jones, M. Schonlau, and W. J. Welch, "Efficient global optimization of expensive black-box functions," *J. Global Optim.*, vol. 13, pp. 455–492, Dec. 1998.

[29] A. Candelieri, "A gentle introduction to bayesian optimization," in *Proc. Winter Simul. Conf. (WSC)*, Phoenix, AZ, USA: IEEE, Dec. 2021, pp. 1–16.

[30] R. Garnett, *Bayesian optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2023.

[31] P. Luong, D. Nguyen, S. Gupta, S. Rana, and S. Venkatesh, "Adaptive cost-aware bayesian optimization," *Knowl.-Based Syst.*, vol. 232, p. 107 481, Feb. 2021.

[32] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. De Freitas, "Taking the human out of the loop: A review of bayesian optimization," *Proc. IEEE*, vol. 104, no. 1, pp. 148–175, Dec. 2016.

[33] J. Wilson, F. Hutter, and M. Deisenroth, "Maximizing acquisition functions for bayesian optimization," in *Adv. Neural Inf. Process. Syst.*, vol. 31, 2018.

[34] A. Makarova, H. Shen, V. Perrone, *et al.*, "Automatic termination for hyperparameter optimization," in *Int. Conf. Mach. Learn.*, PMLR, 2022, pp. 7–1.

[35] "Salome: The open-source integration platform for numerical simulation." Accessed: Mar. 14, 2024, EDF, CEA, and Open Cascade. (2024), [Online]. Available: https://www.salome-platform.org/.

[36] J. Schöberl, "Netgen an advancing front 2d/3d-mesh generator based on abstract rules," *Comput. Vis. Sci.*, vol. 1, no. 1, pp. 41–52, Oct. 1997.

[37] N. Stander and K. J. Craig, "On the robustness of a simple domain reduction scheme for simulation-based optimization," *Eng. Comput.*, vol. 19, no. 4, pp. 431–450, Jun. 2002.

[38] F. Nogueira. "Bayesian optimization: Open source constrained global optimization tool for python." Accessed: Mar. 3, 2025. (2014), [Online]. Available: https://github.com/fmfn/BayesianOptimization.

[39] J. E. Miltat, M. J. Donahue, *et al.*, "Numerical micromagnetics: Finite difference methods," *Handbook of magnetism and advanced magnetic materials*, vol. 2, pp. 742–764, Sep. 2007.

[40] H. Moustafa, A. Kovacs, J. Fischbacher, *et al.*, "Reduced order model for hard magnetic films," *AIP Adv.*, vol. 14, no. 2, Feb. 2024.