Master Thesis

Computational Workflow Optimization for Magnetic Fluctuation Measurements of 3D Nano-Tetrapods

von

Jonathan Pieper

geboren am 11.12.1990 in Marburg

vorgelegt am Fachbereich Physik

der Goethe Universität Frankfurt am Main

Offenbach am Main, Februar, 2021

Reviewers:

Prof. Dr. Jens Müller (Supervisor)
Prof. Dr. Michael Huth

Supplemental Information (see Appendix A):
HTTPS://MASTER.ODY5.DE

ABSTRACT

The detailed understanding of micro–and nanoscale structures, in particular their magnetization dynamics dominates contemporary solid–state physics studies. Most investigations already identified an abundance of phenomena in one–and two–dimensional nanostructures. The following thesis focuses on the magnetic fingerprint of three–dimensional CoFe nano–magnets, specifically the temporal development of their hysteresis loop. These nano–magnets were grown in a tetrahedral pattern on top of a highly susceptible home–build GaAs/AlGaAs micro–Hall sensor using focused electron beam induced deposition (FEBID).

During the measurements, utmost efforts were employed to exemplify current best research practices. The data life cycle of the present thesis is based upon open–source data science tools and packages. Data acquisition and analysis required self–written automated algorithms to handle the extensive quantity of data. Existing instrumental–controlling software was improved, and new Python packages were devised to analyze and visualize the gathered data. The open–source Python data analysis framework (`ana`) was developed to facilitate computational reproducibility. This framework transparently analyses and visualizes the gathered data automatically using Continuous Analysis tools based on GitLab and Continuous Integration. This automatization uses bespoke scripts combined with virtualization tools like Docker to facilitate reproducible and device–independent results.

The hysteresis loops reveal distinct differences in subsequently measured loops with identical initial experimental parameters, originating from the nano–magnet's magnetic noise. This noise amplifies in regions where switching processes occur. In such noise–prone regions, the time–dependent scrutinization reveals presumably thermally induced metastable magnetization states. The frequency–dependent power spectral density uncovers a characteristic $1/f^2$ behavior at noise–prone regions with metastable magnetization states.

CONTENTS

CHAPTER 1
INTRODUCTION

»As students of Physics we observe phenomena under varied circumstances, and endeavour to deduce the laws of their relations. Every natural phenomenon is, to our minds, the result of an infinitely complex system of conditions. What we set ourselves to do is to unravel these conditions, and by viewing the phenomenon in a way which is in itself partial and imperfect, to piece out its features one by one, beginning with that which strikes us first, and thus gradually learning how to look at the whole phenomenon so as to obtain a continually greater degree of clearness and distinctness. In this process, the feature which presents itself most forcibly to the untrained inquirer may not be that which is considered most fundamental by the experienced man of science; for the success of any physical investigation depends on the judicious selection of what is to be observed as of primary importance, combined with a voluntary abstraction of the mind from those features which, however attractive they appear, we are not yet sufficiently advanced in science to investigate with profit.« [1]

James Clerk Maxwell (1870)

Throughout the past century, there has been a rapid development in science and industry. Many communities struggle to keep up with these rapid technical developments since the emergence of computers. The computer has grown a powerful working tool and influenced human society on many levels. As predicted by Feynman [2] and Moore [3], computing power has improved exponentially for over half a century. Irrespective of the approaching end of Moore's law, information technology embraces the possibilities to grow in various other disciplines [4, 5].

Electronics and material science

Condensed–matter physics and material science have contributed during the last decades to this advanced growth with quantum materials [6, 7] that allow novel spintronics devices [8–11], utilizing the electron's spin for semiconductor components with tailored properties. This allows the combination of logic operations and data storage for novel in–memory [12–14] and brain–inspired (neuromorphic) [15, 16] hardware designs. Many recent advancements in information technology are based on such discoveries. The resolution of detectors has significantly improved primarily by component miniaturization down to the micro–and nanoscale [17]. At these scales, quantum phenomena emerge, such as the wave–particle duality or the uncertainty principle. Investigating these and related phenomena led to considerable progress in nanomagnetic material research [18], notably two–dimensional materials [19] and three–dimensional nano–magnetism [20]. These nanomagnetic materials contribute to the detailed understanding of the magnetization dynamics [21] of fundamental magnetic structures, like domain walls, vortices, skyrmions, and magnetic monopoles [22–25]. Material specific properties are often too complex to find analytical or even numerical solutions for problems, like identifying the phase diagram [26]. Instead, such properties can be detected through a comprehensive investigation of novel materials and structures in various environments. The investigation of such novel materials and structures requires the development of high–resolution experiments and simulations. Both endeavours rely highly on efficient computers and algorithms.

This Study

I developed a novel method to measure magnetic fluctuations and analyze them using a self–written object–oriented Python programming framework *ana*. A programming framework is an implemented collection of algorithms to provide a stable code for consistency and allows users to focus on areas of expertise. Object–oriented frameworks can build on class hierarchies, inheritance, and encapsulation to increase extension and re–use opportunities [27, 28]. `ana` is such a framework inspired by CERN's ROOT data analysis framework [29]. `ana` is necessary to handle the data of over 500 recorded magnetic measurements and allows easy access to various measurement details, analyses, and visualizations. `ana` is designed as free and open software for best re–use and reproduction of results. This goal can be accomplished with state–of–the–art data science tools and methodologies.

1.1 Computational Research

Data–driven research

Data science is a relatively new and interdisciplinary scientific discipline [30, 31] and is currently promoted by publishers [32–34]. One goal is to improve scientific research using computational skills with a focus on reproducibility and reusability. Many computational disciplines facilitate open science and open data[1] [35–37] to increase

[1] See also »open« definition in Nomenclature.

productivity and collaboration [38]. Data–driven research focuses on every stage of the data life cycle [30, 39], including acquiring, maintaining, processing, analyzing, and communicating the data. Because of the importance of high–quality software and data in modern research and, in the self–reliance of support, computational best practices are an evolving field and deserve a short review [40–44].

Box 1 | Best Practices for Reproducible and Reusable Research

Principles Include data, algorithms, or other central or integral information in the publication. If this is not possible, it should be made freely accessible through other means [42].

Workflow Tracking and documenting the workflow is vital to enable reproducibility and re–use by others [42].

Data All data should be *findable, accessible, interoperable, and reusable* (*FAIR*) [45–47]. The original data should be kept intact, version–controlled, enriched with metadata, and shared with a permanent identifier [41, 48–50].

Code and Methods Source code must be available and accessible using version control [44, 51–54]. All results should include input values and other parameters with code and scripts [42].

Guidelines All researchers should follow their associated data and code sharing guidelines [55–59].

Licensing Data and code should always be made available for re–use through open licensing [42, 60–63].

Test and Automate Suitable test functions can automatically test the code using continuous integration services [64, 65].

Credit All contributions to a project should be acknowledged, and data and software should be cited [66, 67]. Code can and should be re–used and adapted under an Open Source Initiative (OSI) approved license [42, 60, 62, 63, 68].

Data in science

Data is the source of every scientific discovery, and its quality determines the power and success of an investigation. As stated by Maxwell above, researchers focus on finding striking evidence in gathered data. In this process, a mindful selection of data is necessary to convince the scientific community successfully. However, an equal spotlight should be directed to the *findability, accessibility, interoperability, and reusability (FAIR)* guiding principles for scientific data management [45–47, 69]. These principles realize good research practices [55–57, 59]. Many universities [70–72] and networks [36, 38, 48–50, 73–76] have already adapted the FAIR principles into their scientific routine, and assessing methodologies have been developed [77–79]. Data transparency holds excellent prospects in science [80–82] and positively impacts publications' success [83]. A recent publication [84] shows a statistically significant amount of phosphine gas on venus (implicating possible extraterrestrial life), which could not be reproduced [85, 86]. This additionally highlights the importance of data availability and reproducibility in modern research. A selection of guides [31, 41, 87, 88] and examples [40, 76, 89–91] emphasize the importance of data availability. The fact that even the slow grinding mills of politics nowadays publicizes an intensified interest in FAIR principles [48–50, 92–99] ultimately highlights their prominence.

Reproducible research

A debate has gained interest throughout the last years concerning a *reproducibility crisis* [100–105]. While mainly concerned with medical and social sciences [106, 107] or statistical misinterpretations [108–113], all computational sciences [114] are affected. The problem's primary source is missing documentation and unmaintained code, which leads to difficulties when hard–or software gets out of date [64, 115, 116]. Computational sciences made various contributions to tackle this problem introducing workflows [38, 117] and guides [39, 42, 43, 72, 118–120] for best practices. Interested readers are referred to additional literature [121–128] and Box 1.

1.1.1 Open Source Tools

Data science with Python

Python is an object–oriented interpreter based programming language and an ideal tool for scientists [129–132]. It combines high–level flexibility and readability with low–level capabilities, like linking to third–party libraries in C, C++, or Fortran [133, 134]. The open–source community has contributed many Python modules to ease access to these low–level high–efficiency libraries [135–141]. The Python documentation generator Sphinx [142] can combine docstrings from Python packages with other reStructuredText or markdown files and convert them into a user–readable format. This tool documents the data and code *workflow* for distribution as a web page or printed document. In large software projects, it is common for the lines of documentation to exceed the code lines. A complete list of the used tools is available in Appendix B.

Continuous Analysis

GitLab [143] provides a single open–source tool to version control files with vast adaptability for collaboration. It is similar to the popular GitHub [144] and builds on the popular decentralized version control software git [145–147]. Comparable to GitHub Actions, GitLab is equipped with *Continuous Integration / Continuous Development (CI/CD)* [148] tools for the entire »DevOps« life cycle, which can be perfectly integrated with Python [149]. Container technologies like Docker [51, 150] allow creating virtual environments with defined software versions that perform in a repeatable universal way. Uniting Docker containers with CI/CD allows the automatic re–execution in a well–defined environment [151, 152]. »*Continuous Analysis*« describes a workflow that applies this combination to rerun a computational summary automatically after data or code has been updated [117, 153].

Legal framework

Lastly, trying to publish the data and source code naturally raises questions regarding licensing [60]. From the legal perspective, every creative work is protected by the German »Urheberrecht,« which includes special treatment to source code [154].

Raw measurement data do not apply to this law; however, an original selection and arrangement of the data can be licensed as a collection. Such collections and more creative works other than programming code can be comfortably shared publicly. For such a purpose, the nonprofit organization »Creative Commons« provides various licenses and tools that quickly grant copyright permissions for such creative works [155]. On the other hand, software and source code can be shared and protected amidst multiple terms of conditions and limitations [62, 63]. The literature argues that a copyleft (share–alike) concept shall be inappropriate for the scientific context [60]. Nevertheless, I decided for personal software projects to publish small scripts using the MIT license [156] and preserve larger projects by exercising the obligations of the GNU General Public License (GPL) [157, 158]. The GNU GPL protects the rights of software users by

a) giving them the irrevocable rights of usage, modification, and redistribution of the software; and
b) requiring developers on redistribution to provide the full source code and documentation of changes and functionality;

while legally supporting software developers through patent and warranty clauses. The included obligations preserve openness and assist collaboration. Appendix B highlights some permissive and share–alike licenses. Such legal actions are necessary to grant interested readers the rights to scrutinize an investigations' findings. The findings of the following study are available via the supplemental information (see Appendix A) and concern the magnetic responses' scrutinization of three–dimensional nano–tetrapods.

1.2 Nanomagnetism

Low–dimensional nano-structures

Geometric restraints can group nanomagnetic systems into different dimensions. Single domain nanoparticles (zero–dimensional) [159] are promising candidates for biomedicine [160] and spin–transfer devices [161, 162]. Magnetic nanowires (one–dimensional) can be applied for racetrack–memory [163, 164] and logic devices [165]. Gaining interest emerged for the topologically interesting thin films (two–dimensional) with observations of complex magnetization states [19, 166, 167]. These new phenomena are investigated using multi thin film layers, heterostructures, and magnetic tunnel junctions [168, 169].

Spin–Ice systems

Particular two–dimensional lattices, like a squared or kagome lattice, triggered magnetic frustration effects [170], the geometrically enforced prevention of order, which results in residual entropy at vanishing temperature. Similar behavior is observed in water ice [171], which led to the new class's name: *spin–ice* systems [172]. Customized fabrication further expanded the number of scrutinize–able systems

[173, 174]. Advancements towards *three–dimensional nanostructures* [175] open the opportunity to investigate several interesting geometries, including the pyrochlore lattice, a diamond–like lattice. For example, such lattices allow the examination of magnetic monopoles [176–178].

Fabrication techniques

In the center of this research are such artificial three–dimensional nano–magnets. There are numerous methods available to grow such structures [175, 179]. This study concentrates on magnetic nano–tetrapods grown by means of *focused electron beam induced deposition* (*FEBID*) [180–183] in the group of Prof. Michael Huth (Goethe University Frankfurt) [184–187]. This technique facilitates various geometries, materials, and tailored characteristics [168, 188–191], making artificial spin ice structures [192, 193] and other functional electronic components like Hall sensors [194, 195] possible. Diverse techniques involve photon– or ion–based deposition methods [194–197]. The first–generation of the investigated nano–tetrapods created by FEBID is well documented in the literature [198, 199]. Early studies of this first–generation found a strong magnetic coupling between different magnetic nano–tetrapods [192, 200–202]. Such strongly correlated systems exhibit phase transitions, which can be explored through numerous methods [203].

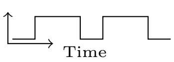
Magnetic measurements

Resolving small magnetic moments with a high resolution is a difficult task. Most techniques involve the sensitive detection of magnetic stray–fields, reflecting the total magnetization of nearby structures. Superconducting quantum interference devices (SQUIDs) [204] and Hall devices [205–207] are susceptible tools for such measurements. SQUIDs are more sensitive than Hall sensors [208] but can not be employed in similar flexible temperature ranges. Hall sensors can be fabricated of different materials for several purposes [194, 195, 209]. This study concentrates on stray–field studies using a GaAs/AlGaAs *micro–Hall magnetometer* [210–212] (see Chapter 3). Such devices form a two–dimensional electron gas (2DEG) with high mobilities of electrons at low temperatures [213, Ch. 10.2.]. This makes them ideal candidates for the sensitive detection of magnetic stray–fields at low temperatures [214, 215].

Noise

Despite being often seen as an unwanted relict, noise can contain valuable information about a system [216–218]. It is usually investigated employing *fluctuation spectroscopy*, transforming the signal into the frequency domain using a Fourier transformation [219]. Electronic noise comprises various sources, like magnetic domains and spin fluctuations, charge carriers crossing an energy barrier, or other material effects [220]. All electronic resistors at finite temperatures exhibit thermal noise, which was first observed and described by Johnson [221] and Nyquist [222]; and can be utilized as a thermometer [223, 224]. Together with the current dependent shot noise (after W. Schottky) [225] and quantum shot noise [226], this represents the class of »white« noise, which is frequency–independent up to a specific cutoff frequency. On the other side, a random telegraph signal (see margin) leads to a Lorentzian spectrum ($S \sim 1/f^2$) with a characteristic corner frequency

Random telegraph signal:



(*Generation–Recombination–Noise*). Theoretically, this is described by two–level fluctuation processes with different time–constants [227]. In nature, a universal $1/f$ *noise* behavior is observed, often described as a superposition of these processes, dominating at low frequencies [228–230]. All noise contributions mentioned above cumulate in laboratory conditions and deserve a closer inspection.

Sensor noise

Studies over the past decades have provided crucial information about noise in semiconducting devices [231–233]. Previous research has established that noise in semiconducting Hall sensors and SQUIDs scales with the external field [234–236]. In Hall sensors, the noise can additionally be used to determine the sensitivity [237]. In micron–and sub–micron–sized GaAs/AlGaAs Hall devices, the aforementioned $1/f$ noise dominates at low frequencies [211, 212, 238–240].

Magnetic flux noise

Magnetic noise reveals more information about magnetization dynamics [241–244]. In particular, theoretical calculations indicate that magnetic monopole noise is detectable in artificial spin ice systems [245, 246]. There are relatively few historical studies in the area of *magnetic flux noise* (*MFN*). Most previous studies focused on detecting MFN in SQUIDs [236, 247, 248] or other superconductors [249–251]. Further investigations on magnetic Barkhausen noise revealed a characteristic $1/f^2$ behavior that this study aims to reproduce and scrutinize [252–256].

Thesis content

This study offers insight into such fluctuation measurements of the measured Hall signal. Further scrutinization of the time–signal at static states inside the hysteresis reveals spontaneous switching processes, indicating *metastable magnetization states* (see Chapter 4). This is the main incentive of combining the two established preceding measurement techniques, micro–Hall magnetometry and fluctuation spectroscopy, to scrutinize the magnetic flux noise of three–dimensional nano–tetrapods. The following thesis focuses on the challenges to perform each step of the research process *lege artis* [55, Guideline 7], explaining the continuous quality assurance mechanisms taken in maintaining, handling, and documenting scientific data. Figure 1.1 highlights the primary topics of each part. Chapter 2 outlines the design of EVE and `ana` as main software tools, following by the continuous analysis workflow to integrate these tools into the daily routine (left). The experimental measurement details (right) are explained in Chapter 3. Chapter 4 presents the analyzed results, which are subsequently discussed in Chapter 5. Closing suggestions for future endeavours are provided in Chapter 5.1. Chapter 6 summarizes key points and conclusions.

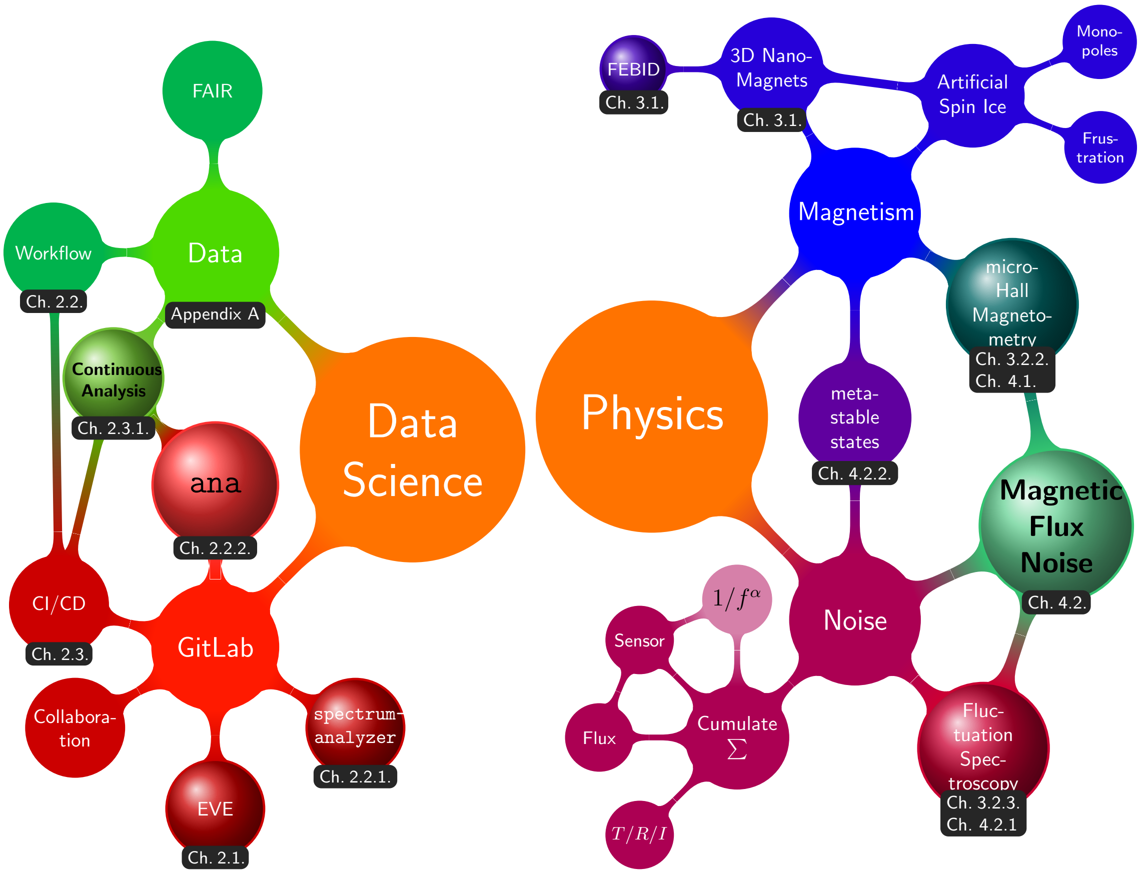


Figure 1.1.: **Mindmap structure of this thesis**. The following thesis utilizes the data science toolbox (left) to investigate the noisy magnetic characteristics of three–dimensional nano–tetrapods (right). Balls are highlighting programmed software tools (left) and experimental methods (right). Black annotation boxes below major topics refer to the corresponding sections in this thesis. The study's primary focus is emphasized in black font.

CHAPTER 2

CONTINUOUS ANALYSIS

Brief Summary

- I updated and improved the essential software to measure, control, and acquire data (**EVE**).
- A novel Python data analysis framework (`ana`) calculates and visualizes measurements.
- **GitLab** introduces a server–based distributed version control and documentation platform.
- **Docker** allows virtualization and well–defined software versions.
- **Continuous Analysis** combines Docker with GitLab Continuous Integration (CI) for automated analysis and better reproducibility.

Data are vital facts for every scientific discovery and need special attention. Data are becoming more decisive nowadays with cheap storage and novel cloud computing capabilities that rely on big data, collecting and combining data on a colossal scale. The following chapter concentrates on the challenges of the data life cycle. First, the data acquisition introduces EVE (short for Efficient Virtual Environment), the primary measurement program used to control the instruments. I enhanced, updated, and documented EVE intensively during this thesis. The enhancements enable unused functions in existing instruments and increase stability and debug–ability. The data, programming code, and corresponding documentation are stored on a self–maintained GitLab server. Data analysis is based on the self–programmed Python data analysis framework (`ana`) and automated for repeatability utilizing Continuous Integration (CI) and Docker.

2.1 Data Acquisition

Data acquisition (DAQ) is the sampling of a signal, typically digitalizing it for post–processing. Here, this task is performed by the group–internal measurement program EVE. It communicates with the instruments to command and read the present state. It reads each selected instrument's currently observed value in pre–defined time intervals and saves it into a comma–separated–value (csv) file.

2.1.1 Established Tools

EVE EVE was programmed in collaboration by this working group's students over the last years and continuously extended [257–261]. The graphical user interface (GUI) of EVE is based on the popular PyQt project and supports adding multiple instruments to a measurement process. Each instrument provides its own GUI, which is created when it is added to the instrument list. An instrument can be added multiple times to command various devices of the same type.

Flowchart Figure 2.1 displays a schematic view of EVE's internal design. The yellow arrows indicate function calls by another part of the program, while the green arrows indicate data transfers, e.g., writing data into a variable or file. The upper half shows the main program, which consists of multiple classes and threads. The main program connects all buttons and visual inputs with the corresponding classes and creates an instance for each initialized class. Some classes inherit threading capabilities from PyQt5's `QThread`. Threading allows Python to break out of the sequential execution and jump between different threads. With today's multi–processing capabilities, this threading approach can be improved with Python's similar core–library `multiprocessing`. These threading and multi–processing capabilities allow the integration of various instruments.

Instruments While EVE can handle multiple different instruments, most user modifications are limited to an individual instrument. The lower half of the flowchart shows a single instrument class with the required functions and procedures. This system of separate modulated instruments allows users to easily modify and add new instruments without understanding all programming backgrounds of EVE. Each instrument class serves a tailored GUI that is embedded into the instruments tab of EVE. If available, most instruments also provide an Auto GUI (right) that allows automated measurement routines. A routine consists of two matrices (right–top) that represent a list of scheduled commands. Such a routine can be easily programmed with EVE and saved as a text file for re–use. Unfortunately, EVE 3.0 alpha was unable to read previously saved routines. This problem could be solved over time, but not all used routines have been saved with the measurements.

2.1.2 Enhancements

Updates EVE was initially programmed in Python 2 using PyQt4 and other packages from

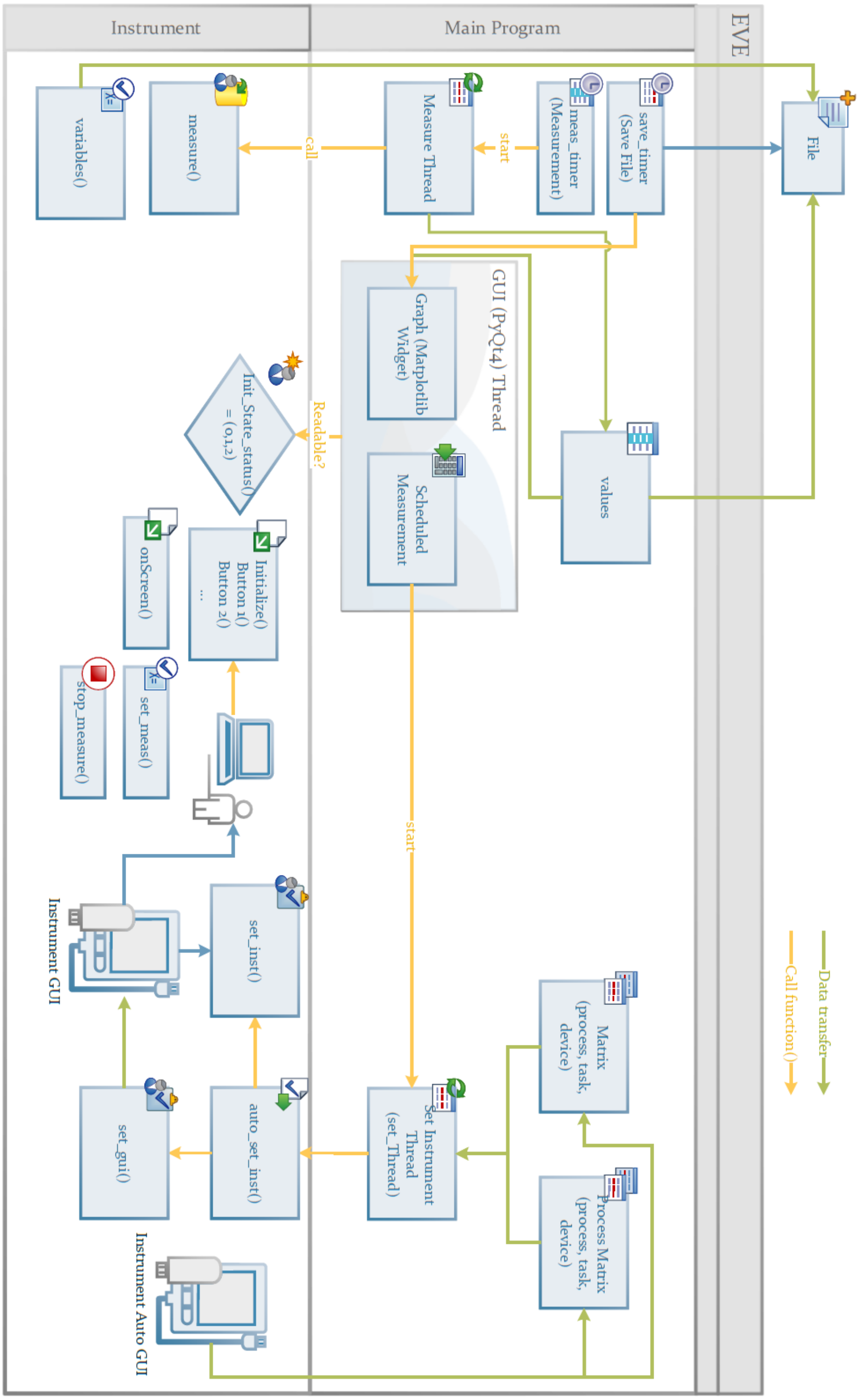


Figure 2.1.: **Functional flowchart diagram of EVE's fundamental objects**. The main program (top) consists of multiple different threads. Each instrument class (bottom) is initialized as a single thread.

the outdated Python(x,y) project[1]. This project has been discontinued and is not updated anymore. Therefore, to update EVE with new features and include recent packages, the code needed to be upgraded. This is especially challenging when facing renamed classes and functions in the Python packages PyQt5, pyvisa, and matplotlib. PyQt5 additionally changed some inheritances and moved functionalities between main modules. To fully document the code changes made, I installed a local GitLab server providing version control for EVE since version 2.6.3. GitLab [143] implements git [145] and makes details findable through a user–friendly web interface. I also ensure an easy installation of EVE through Anaconda. Together with Anaconda, GitLab provides easy access to all EVE versions with the needed software requirements. Anaconda maintains various separated environments with a package manager. The package manager determines all dependencies when installing new software. This ensures the stability of the interplay between different packages. Anaconda's environments allow a more straightforward setup for EVE.

Functional updates

Further EVE enhancements developed throughout this study include functional updates. The Python core–library `logging` improves the debugging process by saving relevant debug messages into a log file. This improves the debugging process and increases developing time. Optionally added command line arguments[2] influence the number of log messages. Multiple new instruments were added, including `HF2LI` (high–frequency lock–in amplifier, Zurich Instruments), `MFIA` (impedance analyzer, Zurich Instruments), and `LS336` (temperature controller, Lake Shore). Additionally, data acquisition functionalities of other instruments were improved.

Data Acquisition (DAQ) Instruments

NI PCI–6281

First data acquisition experiments were performed with a PCI–6281 (National Instruments, NI) data acquisition card. It is directly connected to the Peripheral Component Interconnect (PCI) bus connection on the computer and provides an application programming interface (API) through the included NI–DAQmx driver. In previous experiments, other group members have programmed this instrument in EVE to measure the raw time–signal and calculate the power spectral density (PSD) [257–259, 262]. The PSD is calculated through these digital algorithms that split, filter, normalize, Fourier transform, and scale the signal. I extracted these calculation algorithms from EVE into a new Python module called `spectrumanalyzer` to make it available for re–use. Additionally, the plotting library slowed down the PSD calculation. I solved this issue with a faster plotting algorithm (`pyqtgraph`). A second instrument in usage is also capable of recording raw time–signals.

SR830DAQ

The lock–in SR830 (Stanford Research Systems) has an internal data buffer storage.

[1] Python(x,y) is a deprecated collection of scientific tools for data analysis and data visualization. See `https://python-xy.github.io/` for further information.

[2] For example: `python EVE.py -d` activates the debug mode and logs every debug message.

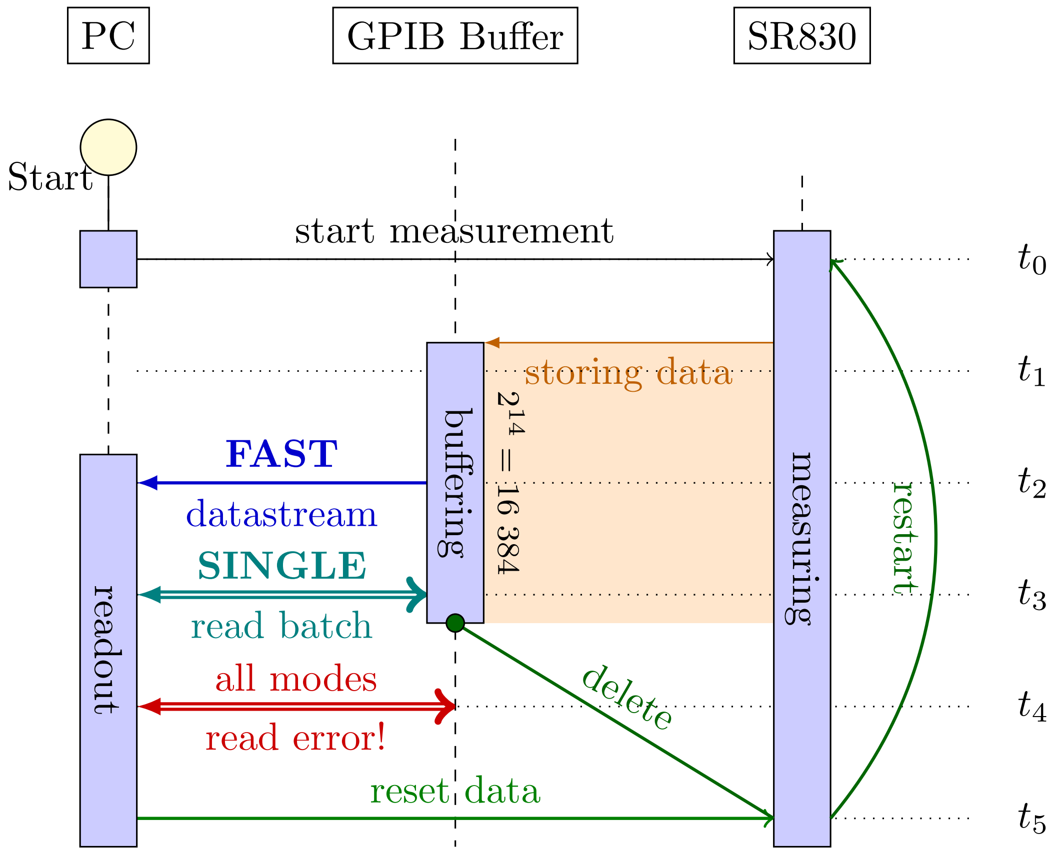


Figure 2.2.: **SR830DAQ Time–stream visualization**. The arbitrary points in time t_0 to t_5 are used to visualize the communication paths between the computer and lock–in amplifier SR830 through the GPIB connection.

This permits measuring the data at a sampling rate of 62.5 mHz $\leq f_S \leq$ 512 Hz. I programmed the corresponding DAQ functionality for the EVE instrument `SR830` to access and utilize this data buffer via the GPIB connection. This new SR830DAQ function provides three different acquisition modes, FAST, SINGLE, and LOOP. Figure 4.8 shows a time–stream visualization. On the right side is an arbitrary time–axis to illustrate the passing time. Triggered by the start button, at the time t_0, the computer sends a command to the lock–in, which starts the DAQ measurement. The lock–in then waits a short time before storing the incoming signal in the buffer at time t_1. The data limit is defined by the GPIB protocol, which allows storing up to 16 384 (2^{14}) values in the buffer. When using the FAST mode, the lock–in automatically sends a large datastream (blue) to the computer at all times t_2 between t_1 and t_3. SR830DAQ FAST mode reads these values simultaneously and stores them in a local `pandas.DataFrame` until stopped. When using SINGLE mode (teal), the lock–in waits a pre–defined time after t_1 and reads out all values in one batch. The third mode, LOOP, repeats multiple SINGLE modes until stopped. At some time t_3, when the GPIB buffer can not save more data, the lock–in is unable to store more data and consequentially stops the measurement. The orange area indicates the stored data. If some mode wants to read the buffered values at time t_4 after the measurement stopped, it receives a timeout error (red) and stops automatically. The only way to get the measurement starting again is to send a reset and start signal at a time t_5, which deletes the buffer to restart the measurement at t_1. The algorithm does this in the LOOP mode after each SINGLE measurement and the FAST mode after gathering 15 000 points. Tests have shown that the LOOP mode loses data points during the long data batch's readout and reset (see description in EVE wiki available via supplemental information in Appendix A).

Data Documentation

Workflow context

All methods mentioned above gather and store the data on a local machine. These data are neither linked nor do they provide any context yet. The experimenter can contextualize these data, ideally providing version–control and metadata. All performed measurements shown in this thesis are documented with parameters, notes, and plots in a OneNote notebook. For more accessible context, the measured data has been version–controlled and augmented with metadata. Each filename consists of the measurement number together with basic parameter settings. These pieces of information are accessible through regular expressions (Regex). A Python script extracts the information and saves it into a csv file. This step is relevant for findability and reusability.

Supplemental information

An evolving effort for better documentation has led towards an open–source driven and markdown based workflow that can convert documentation and metadata into various presentable formats. Additional supplemental information on the data and code has been made available via a web–page. This web–page provides more context for better interoperability. Unfortunately, OneNote does not support an open data format or easy export of the content, and it converts every mathematical formula into a picture. The notebooks have been made accessible by converting them into ReStructured Text (rst) files using Pandoc [263], a universal document converter. The source code and acquired data are documented as well as possible, spending a reasonable amount of time (see Appendix A).

2.2 Data Analysis

This open data approach is consequentially succeeded by the usage of free software for data analysis. The benefit of free software is the independent verifiability of results by reviewers. Therefore, the whole data processing environment is based solely on free algorithms, and all self–written Python source codes are released as free software.

2.2.1 Spectrumanalyzer

First spectrum

As mentioned above, the `spectrumanalyzer` Python module has been extracted from the EVE instrument `NIPCI6281`. The `SpectrumAnalyzer` class contains all algorithms needed to calculate and save a PSD from a time–signal by means of a fast Fourier transform (FFT). This time–signal $V(t)$ is provided as a single list of values to create a `SpectrumAnalyzer` object. Together with additional required parameters, like the sampling rate f_s or the number of first spectra N, this time–signal is then processed by the generator `cut_timesignal`. This generator, as the name suggests, splits the time–signal into equidistant shorter signals $V_{(n)}(t)$, where $n = 1, 2, \ldots, N$. Each of these short signals is filtered using the `scipy.signal.butter` filter. If requested, the algorithm can downsample the signal, keeping every k^{th} point ($k \in \mathbb{N}$) [257]. Because the SR830DAQ function measures with low sampling rates (compared to the NI PCI–6281), this downsampling is a drawback and had been bypassed. The remaining signal then disposes of the mean value, leaving only fluctuations behind. These fluctuations are multiplied by a Hanning window (`numpy.hanning`), Fourier transformed with `numpy.fft.fft`, and squared. A factor of $2 \cdot (3/8 \cdot \ell \cdot f_s)^{-1}$, where ℓ represents the signal's length, counters the Hanning window's effect on the calculated FFT amplitude [264, pp. 30–35]. Finally, the generator yields the calculated PSD $S_V^{(n)}(f)$ for each signal part via an iterator. This iterator can be used in a `for` loop to access each intermediate PSD and the resulting final PSD $S_V(f) = \frac{1}{N} \sum_{n=1}^{N} S_V^{(n)}(f)$. This resulting final PSD is called the first spectrum.

Second spectrum

The above described first spectrum does not always contain all information about the intrinsic noise. In such cases, higher–order power spectra are needed. Here, the second spectrum is calculated [265] through the temporal development of the time–resolved first spectrum $S_V^{(n)}(f)$ (see Fig. 4.10 bottom left plot). The integral of the first spectrum $P_{(n)}(f_a, f_b) = \int_{f_a}^{f_b} S_V^{(n)}(f) df$ defines the power inside a pre–defined frequency octave $O_k = [f_a, f_b)$. The second spectrum $S_{O_k}^{(2)}(f)$ is then derived through the application of the FFT on $P_{(n)}(f_a, f_b)$. This second spectrum essentially gives rise to the PSD's fluctuations inside a given octave O_k. The newly developed `ana` package uses these described algorithms from the `spectrumanalyzer` module to analyze the data.

2.2.2 Data Analysis Framework (`ana`)

Functional specifications

`ana` is an object–oriented Python framework to analyze measurements. `ana` has created all figures in Chapter 4. Figure 2.3 shows ana's classes and their relationship to each other. Inherited classes, which share most variables and functions, are connected by solid arrows. In contrast, dashed arrows indicate the instancing of a class into an object, enabling execution of class-internal functions. `ana` grants access to single measurements via the `SingleM` and inherited classes. These single measurements are customized to a specific instrument and data structure. Additionally, they incorporate algorithms to read, analyze, and visualize measurements. The high–level API (top) takes advantage of creating multiple single measurement objects. This approach allows the combination of various measurements in a single plot. Figure 2.4 shows this dividing approach to access the high–level visualization API through Jupyterlab, while computing analysis is based on robust open–source algorithms inside the low–level backend. More detailed technical specifications are available in the supplemental information.

`HLoop`

$n_e = \frac{I \cdot B_{ext}}{e \cdot V_H} \approx 1.369 \cdot 10^{11} \frac{1}{cm^2}$
e: electron charge

The `HLoop` class handles lock–in measurements during a sweeping magnetic field. It requires EVE's output file for a specific ordered combination of instruments. This output file contains all information about the measured temperature, magnetic field, and voltages at discrete times during the measurement. `HLoop` distinguishes between parallel and gradiometry measurements (see Section 3.2.2). The magnetic stray–field calculation assumes a constant current of $I = 2.5\,\mu$A and electron density n_e, calculated from the Hall voltage V_H at an external magnetic field of $B_{ext} = 1\,$T applied perpendicular to the Hall sensor's surface ($\theta = 0°$, see Fig. 3.2). When the external field is applied in an angle $\theta = 90°$, parallel to the Hall sensor's surface, three different Hall crosses are measured simultaneously. In these parallel measurements, an empty cross signal is subtracted from the two other measured crosses. This approach eliminates the linear and non–linear Hall background from the sensor [266]. In contrast, gradiometry measurements contain a small linear background that is determined with linear regression using `scipy.stats.linregress` on the negative saturated region. The remaining non–linear background can be eliminated through the difference between two different sweeps in the same field. The discrete down–sweep signal $B^{down}(H_{ext})$ and up–sweep signal $B^{up}(H_{ext})$ are interpolated with `scipy.interpolate.interp1d`. The difference plot subtracts these curves $\Delta B_z(H_{ext}) = \langle B^{down}(H_{ext})\rangle - \langle B^{up}(H_{ext})\rangle$. In this process, information loss is unavoidable.

Fluctuations

Measurements from the signal–analyzer (SR785) are usually two–dimensional (see Section 3.2.3), providing the frequency f and PSD $S_V(f) \sim f^{-\alpha}$. The analysis in the `SA` class is reduced to a logarithmic linear regression $\log y = \alpha \log x + x_0$ for a given frequency range. In this linear regression, x and y denote f and S_V, respectively. The regression range is restricted by default to $20\,\mathrm{mHz} < f < 700\,\mathrm{mHz}$ to avoid fitting unrelated artefacts. This regression outputs the PSD's slope α and amplitude $x_0 = S_V(f = 1\,\mathrm{Hz})$. `RAW` measurements contain the whole time–signal $V(t)$, processed through the `spectrumanalyzer` module described above.

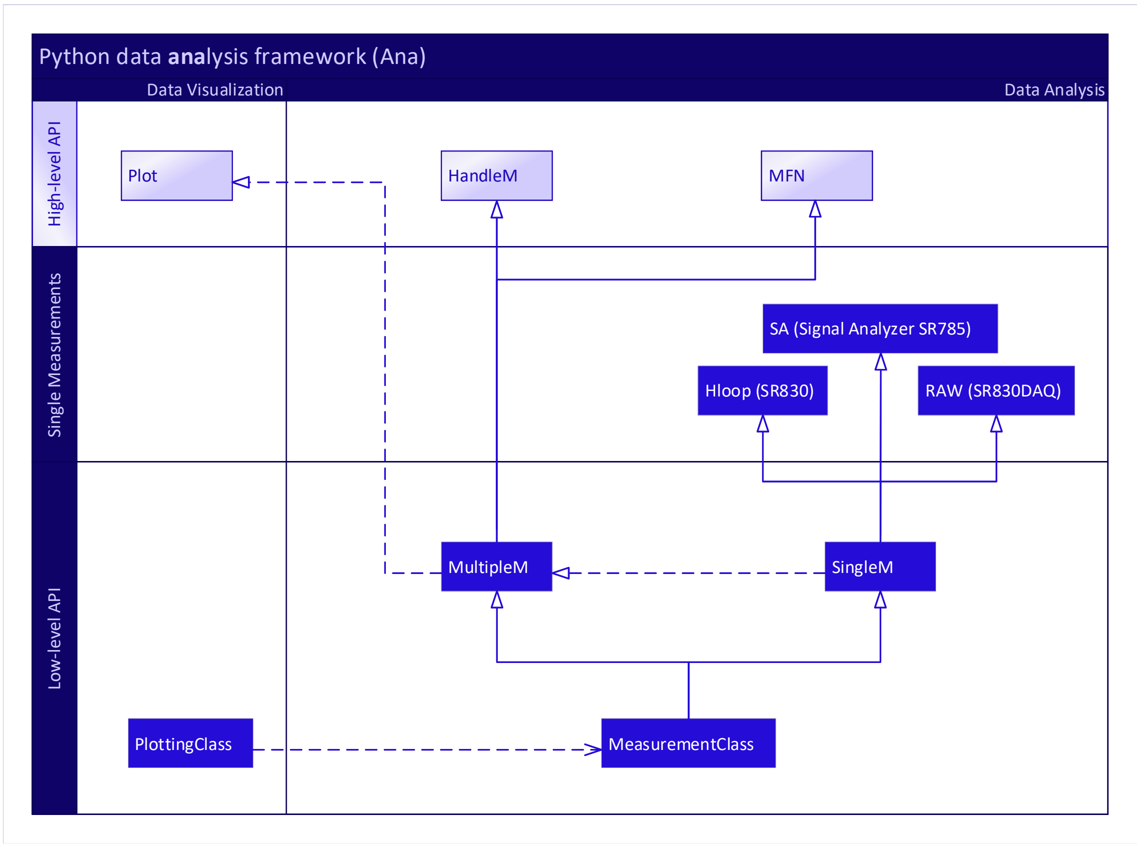


Figure 2.3.: **UML Class–Diagram of `ana`.** This diagram displays the internal relationship between `ana`'s classes using the unified modeling language (UML). Solid arrows indicate a direct inheritance of classes, and dashed arrows show where objects are instanced. `ana` provides a high–level interface to classes that can handle multiple single measurements (shown in the middle) at once. All measurements classes provide additional access to a low–level python backend, which is used by the high–level interface.

A special role in the analysis is the combination of various single measurements. The `MFN` class was created to handle the measurements presented in Section 4.2.2, specifically time–signals during an interrupted field sweep. This class manages multiple `RAW` measurements and analyses automated SR830DAQ routines. These routines create several single measurement files with varying parameters. Essential parameters, like position inside the hysteresis, are encoded into each filename. `MFN`'s constructor function that initializes each instantiation scans all filenames with regular expressions for such parameters and stores them into an `MFN.info` dictionary. Afterward, each time–signal is processed by the `ana.RAW` class, calculating the first spectrum and time–resolved PSDs. Several plotting functions grant visualization of diverse perspectives to the data. A significant plotting function is the `MFN.plot_info()` MFN

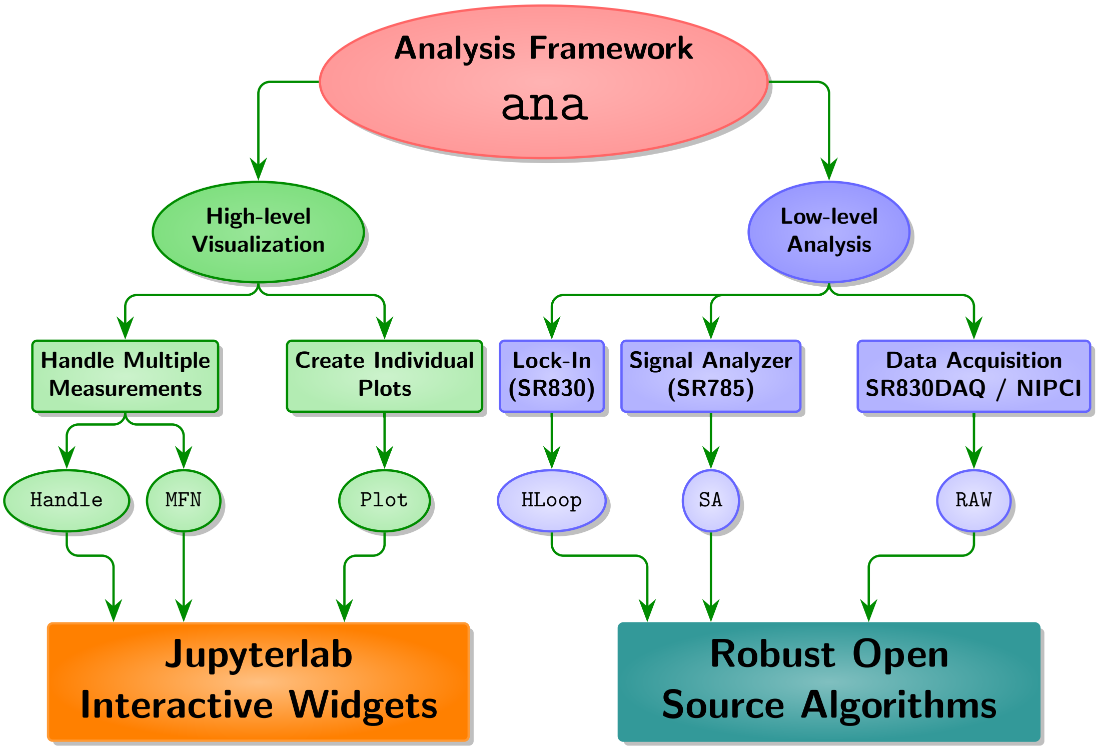


Figure 2.4.: **Hierarchical functional specifications of `ana`.** This tree shows the external interaction possibilities with `ana`. The low–level python backend (right) is based on robust open–source algorithms that link to state–of–the–art C++ libraries. The high–level visualization API allows the interactive exploration of data using widgets.

function that is used to create the analysis overview in Section 4.2.2.

2.2.3 Jupyter Notebooks

`ana` provides the necessary algorithms to analyze and plot the data. These algorithms are then utilized and combined inside Jupyter notebooks. Jupyter serves a powerful integrated development environment (IDE) to write and execute Python code inside a web–browser. Jupyter notebooks are a quick and easy way to navigate data and create plots [267, 268]. The underlying Python kernel facilitates iterative data exploration, enabling to focus on application logic rather than implementation details [129].

2.3 Computational Workflow

Testing

The code described above can read, analyze, and visualize the data. Test cases can additionally improve the stability and reproducibility of critical code segments [269]. I extended `ana` and the `spectrumanalyzer` with test cases that ensure the code stability. The test cases for `ana` also create and save all plots shown in this thesis. The code has been tested on 64bit (x64) Windows and Linux, as well as on arm architectures, like the Raspberry Pi 3 and the new Apple Silicon chip. These tests can also be performed automatically by GitLab CI/CD [270]. Figure 2.5a shows this CI workflow as a flowchart diagram. First, a Docker container needs to be created using a Dockerfile. This container is then uploaded to a registry server, where it is available for other users. The GitLab CI configuration needs to include this Docker container together with instructions. When this configuration file is in place, GitLab CI automatically runs the defined scripts on every push of changed source code or data and creates a pre–defined output.

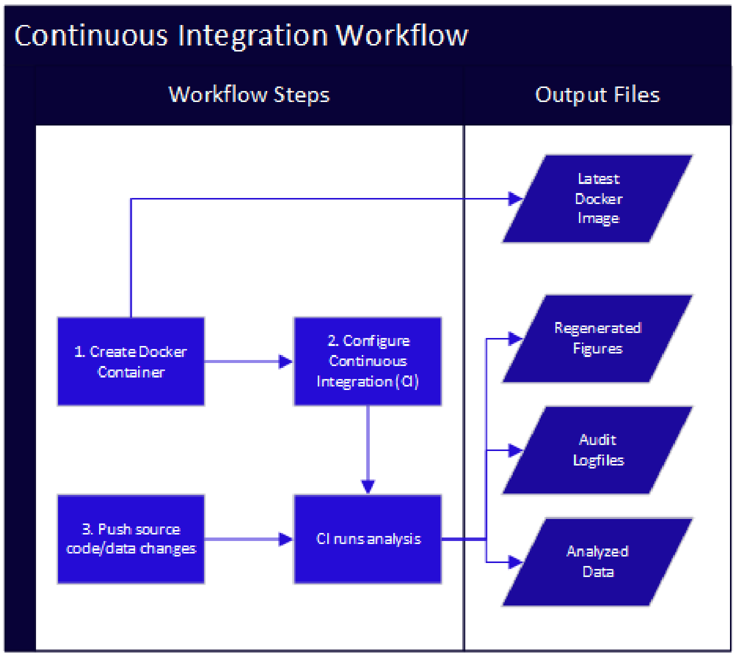
2.3.1 Continuous Analysis

Version control

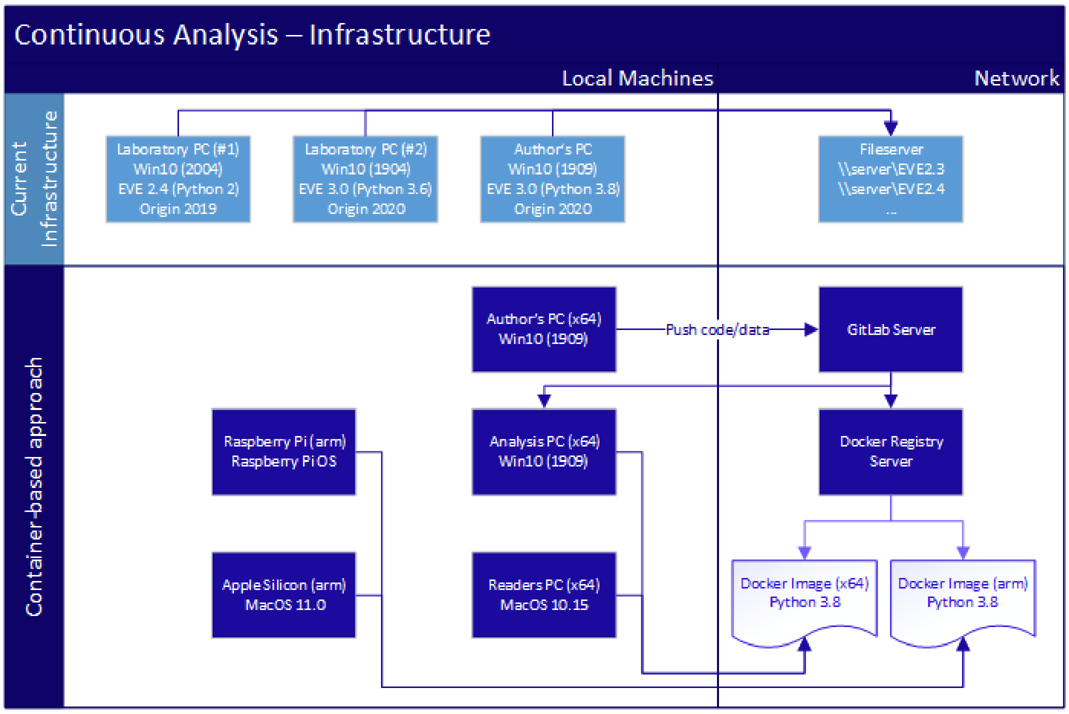
The past decades have seen an enormous change in the software development landscape. Nowadays, it is a well–established philosophy to take advantage of a version management system to track differences and history of changes and support more extensive collaboration. The currently used infrastructure (see the top of Figure 2.5b), where every computer accesses a public folder in the local network, has already proven efficient and successful over the years. Nevertheless, recent developments of novel free software facilitate improving this infrastructure and increasing productivity and collaboration. GitLab provides secure version control of all projects and automatically analyses the data with Continuous Analysis. Multiple tutorials are currently created online [271–273] and published in journals [151, 152, 274–277] to teach scientists programming skills, as well as source–code maintenance and testing with git and CI tools.

GitLab CI

Listing 2.1 shows an annotated excerpt of the used GitLab CI configuration. A GitLab runner, executed as a permanent service on the analysis PC, downloads the git repository from the server and executes scripts as configured. Specific variables, like `GIT_SUBMODULE_STRATEGY`, influence the behavior of the runner. Stages consist of one or more jobs that run in a given order. Line 10 defines the used Docker image for a virtual environment. This Docker image contains all requirements needed to run the code. Line 12 names the current job and all following indented lines! If some packages or requirements are still missing, a `before_script` section executes commands to install them. This installs the local packages `spectrumanalyzer`, `ana`, and EVE. Afterward, the `script` section runs all tests in the given `TEST_DIR`. This variable is changed in lines 36 and 41 to execute the tests from a different test direc-



(a) CI Workflow



(b) Continuous Analysis Infrastructure

Figure 2.5.: **Continuous Analysis**. a) Schematic view of the Continuous Integration workflow and created outputs. b) Depiction of changes in the infrastructure. Currently, all files are saved and accessed through local network folders. Continuous Analysis centralizes data and requirements on the GitLab and Docker registry servers, making data, code, and dependencies easily accessible. Code execution and data analysis can be performed on any computer using the corresponding Docker image. Inspired by Beaulieu–Jones and Greene [153].

tory `TEST_DIR`. The output files of the scripts can be uploaded to the GitLab server as artifacts. The `artifacts` section starting in line 26 configures file–patterns that point to files that are uploaded after the execution. Optional security scans in lines 43ff. use pre–configured GitLab CI security templates. These can extract additional information about used dependencies and license compliances. The computational environment is defined by the used Docker image in line 10. This image can be created with a self–written Dockerfile.

Listing 2.2 shows an excerpt of the Dockerfile used to create the Docker image. It is based on a pre–configured Python 3.8 image. Lines 3ff. download and install the required software to create the documentation and LaTeX plots. Lines 13ff. define needed Python packages for the Python package manager `pip`. At the end of the file, an optional `entrypoint.sh` shell script is copied to the image to start Jupyterlab. Jupyterlab provides a webserver to program Jupyter notebooks easily. This server is not needed when combining the image with GitLab CI. Therefore, the `ENTRYPOINT` in line 32 has been disabled in the utilized image `jupyterlab:ana`. Typically, each processor architecture (x64 or arm) needs a specially designed Docker image. The Docker image presented here is optimized for x64 processors. In order to create a similar Docker image for arm processors, the pandoc installation would need a specialized arm version in line 11. Dockerfile

This Docker image creates custom Docker containers to automate code testing with GitLab CI. On every updated change of the git repository, GitLab CI automatically runs pre–defined tests and provides various details after execution.

```
variables:
  GIT_SUBMODULE_STRATEGY: recursive

stages:
  - doc
  - spectrumanalyzer                              Docker
  - ana                                           image
  - test

image: registry.gitlab.com/ganymede/jupyterlab:ana

ana:prepair:
  stage: ana
  variables:
    TEST_DIR: tests/ana/prepair         install packages
  before_script:
    - cd spectrumanalyzer && python -m pip install -e . && cd ..
    - cd ana && python -m pip install -e . && cd ..
    - cd EVE && python -m pip install -e . && cd ..
    - python -m pip install coverage
    - cd ana
    - mkdir output
  script:
    - coverage run -m unittest discover -s $TEST_DIR    run tests
    - coverage report -m
  artifacts:             upload created files
    paths:
      - output
  only:
    - master
    - develop

ana:fit:
  extends: ana:prepair
  variables:                              repeat with
    TEST_DIR: tests/ana/fit               different test dir

ana:visualize:
  extends: ana:prepair
  variables:
    TEST_DIR: tests/ana/visualize

sast:                                     security scans
  stage: test
  include:
    - template: Security/SAST.gitlab-ci.yml
    - template: Security/Dependency-Scanning.gitlab-ci.yml
    - template: Security/License-Scanning.gitlab-ci.yml
```

Listing 2.1: ana's **GitLab CI Configuration** (gitlab-ci.yml excerpt)

```
FROM python:3.8        re-use preconfigured Python image

RUN curl -sL https://deb.nodesource.com/setup_12.x | bash - && \
    apt-get upgrade -y && \
    apt-get install -y nodejs \        install software
        texlive texlive-science \
        texlive-latex-extra texlive-xetex \
        dvipng man-db cm-super graphviz && \
    rm -rf /var/lib/apt/lists/*

RUN wget https://github.com/jgm/pandoc/releases/download/2.10.1/
   pandoc-2.10.1-1-amd64.deb && dpkg -i pandoc-2.10.1-1-amd64.deb
   && rm -f pandoc-2.10.1-1-amd64.deb

RUN python -m pip install --upgrade pip && \
    python -m pip install --use-feature=2020-resolver --upgrade \
        jupyterlab \
        sphinx \
        numpy \
        scipy \        install Python
        matplotlib \   packages
        pandas \
        seaborn &&\
    jupyter labextension install \
        @jupyter-widgets/jupyterlab-manager \
        @jupyterlab/toc

COPY bin/entrypoint.sh /usr/local/bin/        copy files
COPY config/ /root/.jupyter/

EXPOSE 8888
VOLUME /notebooks
WORKDIR /notebooks
ENTRYPOINT ["entrypoint.sh"]        install startup script
```

Listing 2.2: **Dockerfile Jupyterlab** (excerpt).
Available at `https://gitlab.com/ganymede/jupyterlab`.

CHAPTER 3

MAGNETIC CHARACTERIZATION TECHNIQUES

Brief Summary

- 3D nano–tetrapods are deposited directly on top of the sensor's active area using **focused electron beam induced deposition** (FEBID).
- **Micro–Hall magnetometry** provides a versatile tool for magnetic stray–field and fluctuation measurements.
- **Fluctuation spectroscopy** extracts information hidden within noise.
- A novel method measures **magnetic flux noise** (MFN) by combining micro–Hall magnetometry and fluctuation spectroscopy.

A Hall sensor is a highly sensitive magnetometer to extract magnetic stray–fields. In the present study, I utilize a custom made Hall sensor for low temperature and microscale measurements. This sensor measures the magnetic signal of three–dimensional nano–tetrapods and explores the magnetic characteristics for dynamic conditions. An automated measurement approach allows high–resolution access to a signal. This high–resolution access facilitates the analysis and decomposition of electric and magnetic noise. By analyzing the electric and magnetic noise, I investigate the magnetic characteristics of newly developed nano–tetrapods.

3.1 3D Nano–Tetrapods

Geometry

The examined structures are deposited on top of the Hall sensor's top gate in collaboration with the group of Prof. Michael Huth (Goethe University Frankfurt) using FEBID. The structures are designed as ferromagnetic CoFe tetrapods, a single lattice element from the diamond structure discussed in the introduction. The magnetic moments in such systems can be frustrated [178, 278]. The tetrapods are placed as 2×2 arrays in the center of one of the hall bar's active areas. Figure 3.1

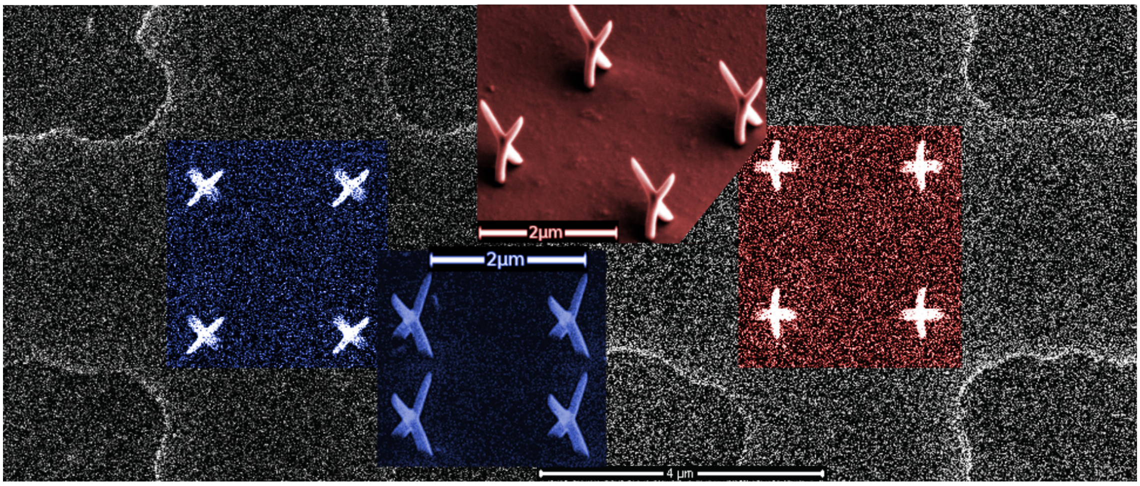


Figure 3.1.: **Hall Sensor with 3D Nano–Tetrapods**. Scanning electron microscopy (SEM) image of the micro–Hall sensor with nano–tetrapods deposited on top. The tetrapods are arranged in 2×2 arrays $2\,\mu$m apart centered in the Hall sensor's active area. This area is highlighted in blue (**Crosses**) and red (**Plusses**). The insets reveal the three–dimensional projection to clarify the arms' relative directions (Images by Fabrizio Porrati).

depicts the used micro–Hall sensor after the deposition of the nano–tetrapods. Because of the tetrapod's characteristic shape from this perspective, the blue (left) and red (right) colored structures are named Crosses and Plusses, respectively. All eight single element tetrapods are identical, differing only in the relative orientation of the external field. The Plusses and Crosses are rotated by 45°, which leads to complex inter–element interactions. The structure's size was chosen for its relatively small and simple geometry. The array consists of four structures per cross. This counters expected difficulties in obtaining a large enough magnetic stray–field from the structures.

3.2 Experiment

3.2.1 Hall Sensor

Sensor A schematic view of the sensor is shown in Figure 3.2. The teal–colored, two–dimensional electron gas (2DEG) is located about 70 nm below the golden top gate. Multiple differently doped $Al_xGa_{1-x}As$ layers are above and below the 2DEG (not displayed in the figure). Such micro–Hall sensors are created with an enormous investment of workforce and specialized instruments [279]. The employed sensor

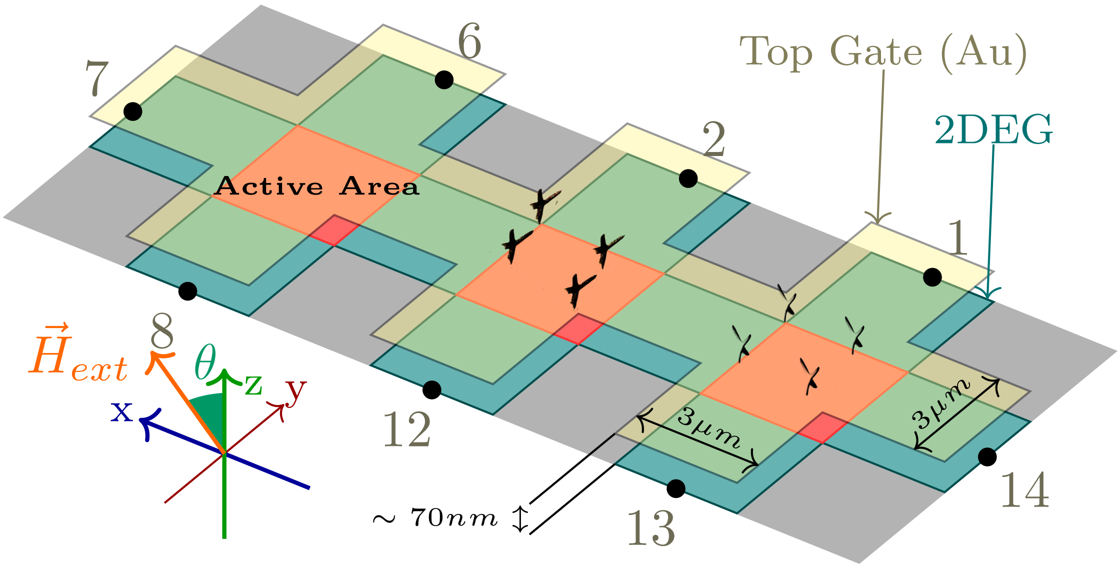


Figure 3.2.: **Geometric sketch of the used Hall sensor**. The gold top gate and conducting two–dimensional electron gas (2DEG) are painted in yellow and teal, respectively. The 2DEG is approximately 70 nm below the top gate and accessible through the numbered connection points. The sensors's active area highlighted red is positioned in the 2DEG and has a size of $3 \times 3\,\mu\text{m}^2$. The nano–tetrapods are deposited on top of the top gate centered in the middle of the active area. The Hall sensor is rotate–able around the y–axis, thus enabling the application of an external magnetic field $\vec{H}_{ext}$ at varying angles θ.

was fabricated by Merlin Pohlit [266] and characterized by Mohanad Al Mamoori [202].

The sensor's top gate serves as a substrate for the nanostructures and allows manipulation of the electron density of the 2DEG. This manipulation reduces avoidable noise sources during the measurement [212]. The Hall sensor's signal–to–noise ratio (SNR $\sim \sqrt{N/\gamma_H}$) depends on the material–specific Hooge–parameter γ_H, and the number of electrons $N = A \cdot n_e$ inside the active area, which depend on the area of the Hall crosses A and the electron density n_e. The top gate and an unused Hall bar connection are connected to the ground in order to avoid potential fluctuations. This grounding is done by the current and voltage source Yokogawa 7651 (see Fig. 3.3b, purple, top right corner). Top gate grounding

(a) Cryostat and IVC (b) Instruments

Figure 3.3.: **Photographs of the experimental setup**. (a) Cryostat (right) with the opened inner vacuum chamber (IVC, left). Insets are showing a zoomed–in view of the rotate–able sample holder (bottom) and Hall sensor. (b) Used instruments for magnetic fluctuation measurements (magnetic power supply and temperature controller not shown). The IVC connection box (yellow) is connected to the 2DEG through the sample holder.

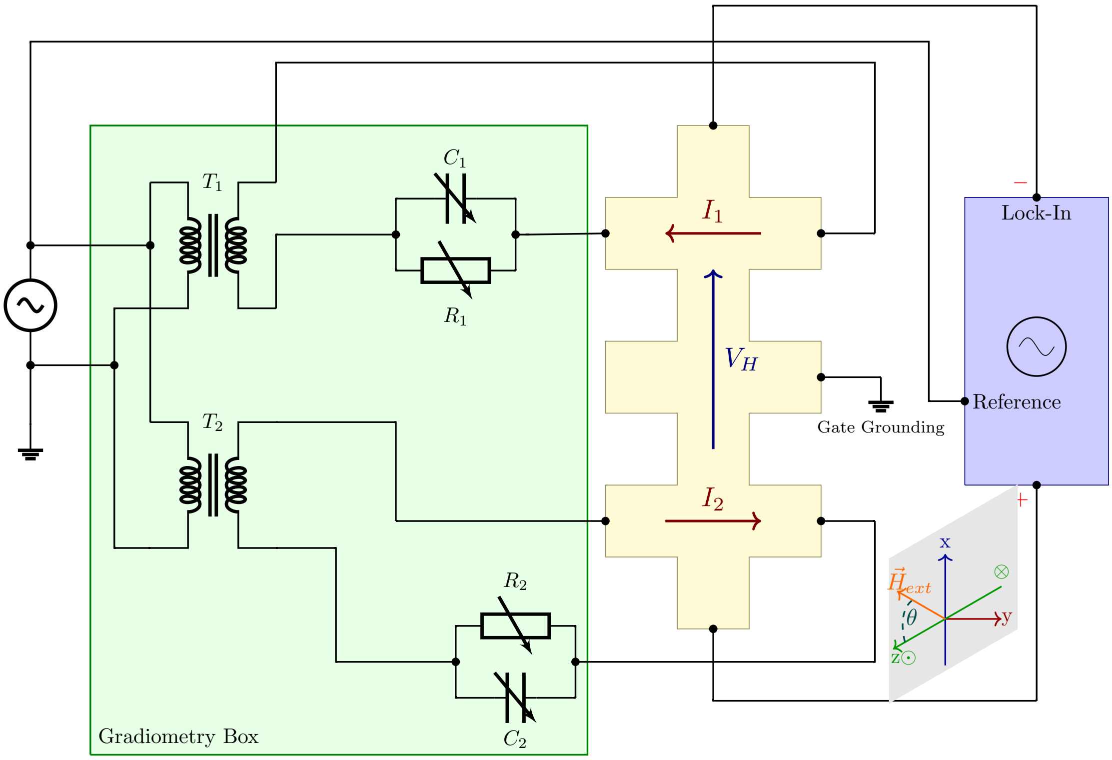


Figure 3.4.: **Gradiometry Setup**. A schematic representation of the experimental setup is drawn as an electric circuit diagram. The external magnetic field $\vec{H}_{ext}$ is applied at an angle θ in the x–z–plane. The gradiometry box (green) splits the voltage into two currents I_1 and I_2. These are sent in opposite directions to cancel the effect of the external field in–situ. The resistors R_1 and R_2 balance the currents while the capacitors C_1 and C_2 eliminate possible phase differences. The Hall voltage V_H is measured perpendicular to the currents, here along the x–axis. The top gate and one vacant Hall cross are grounded in order to suppress potential fluctuations.

3.2.2 Micro–Hall Magnetometry

All experiments were performed inside a helium–4 bath cryostat by Janis Research. Figure 3.3a shows a photograph of the cryostat (right). The cryostat is equipped with a superconducting magnet and an inner vacuum chamber (IVC, left). Inside this IVC, the cryostat provides wiring to a heater and a sample holder underneath the top–most sensor (inset bottom). The sample holder can be rotated around one axis at an angle θ of circa $-120° \lesssim \theta \lesssim 120°$. A small amount of helium exchange gas couples the sample holder thermally to the surrounding He bath. The heater manipulates the temperature during the measurements and is commonly Cryostat operated by a proportional–integral–derivative (PID) temperature controller. The present study uses the temperature controller LS340 (Lake Shore) and the magnet's intelligent power supply IPS120 (Stanford Research Systems) to control the heater and superconducting magnet, respectively.

Hall effect An external magnetic field $\vec{H}_{ext}$, driven through the cryostat's superconducting magnet, changes the magnetization of a placed sample. The magnetized sample creates a local magnetic stray–field, which superposes the external field. This local magnetic stray–field usually allows the observation of a hysteresis loop — the ferromagnetic fingerprint [280]. Hysteresis loops usually depend on the temporal history of the external magnetic field. The resulting Hall voltage $V_H = (n_e \cdot e)^{-1}\langle B_z\rangle$ (e: elementary charge) is proportional to the averaged magnetic stray–field perpendicular to the sensor's surface $\langle B_z\rangle$ (here: z–direction). The influence of the external field on the resulting signal depends on the applied angle $\langle B_z\rangle = |\mu_0 \vec{H}_{ext}| \cos\theta$. In practice, this allows for a precise rotation of the sensor. After determination of the maximum signal $V_{max} = V_H(\theta = 0°)$ for a stationary field, the desired signal can be calculated for any new angle $V_H(\theta_{new}) = V_{max} \cdot \cos(\theta_{new})$. During the rotation, capacitive and thermal effects can influence the result and add an error of approximately $\Delta\theta \approx \pm\, 2 - 5°$.

Experiment Figure 3.3b displays a photograph of the instrument rack highlighting the used instruments. The IVC connection box (yellow) provides wiring to access the sensor inside the cryostat. The default measurement setup for all angles $\theta \neq \pm 90°$ is shown in Figure 3.4. The lock–in amplifier SR830 (blue, Stanford Research Systems) generates an oscillating voltage V_{out} and measures the signal. The gradiometry box (green) splits the oscillating voltage into two separate signals. Here, transformers T_1 and T_2 split the signal before the conductors C_1 and C_2 and resistors R_1 and R_2 are modulating the phase and adjust the resulting current $I_x = V_{out}/R_x$ (where $x \in \{1, 2\}$), respectively. This process results in two currents I_1 and I_2, that flow into opposite directions, for example connecting I_1 (13^+—1^-) and I_2 (6^+—8^-) to measure the Plusses with the gradiometry technique (see Fig. 3.2). The second current I_2, which flows through an empty cross, creates an opposite Hall voltage annulling this superposed signal. The resulting Hall voltage V_H, measured at 7^+—14^-, is reduced by a factor of up to 10^3. The gradiometry technique allows the cancellation of the external magnetic field signal in–situ. This in–situ cancellation was derived from the gravity gradiometry technique [281]. The gradiometry technique improves the sensitivity significantly and provides a powerful tool when facing large external signals.

Background As mentioned above, the gradiometry technique is well suited to remove the background for angles $\theta \neq \pm 90°$. After the in–situ subtraction of the external magnetic field, a linear and non–linear background remains. A linear regression at negative saturation determines the linear part. Determining the non–linear part is more challenging and transgresses the scope of this study. When the external field is applied parallel to the sensor's surface ($\theta = \pm 90°$), the external field superposes no significant Hall voltage. These angles allow sending a single current along with the long bar 7^+—14^- and measuring all three Hall signals simultaneously (in parallel). In contrast to the gradiometry technique, this is called the parallel technique. The small background signal is then obtained from the empty cross. All fitted curves presented in the results are retrieved by subtracting the above–described background. Additionally, a difference plot also eliminates the non–linear background. However, eliminating this non–linear background causes information loss. The difference plot represents the width of the magnetic hysteresis loop at a given field by subtracting them $\Delta B_z(H_{ext}) = \langle B^{down}(H_{ext})\rangle - \langle B^{up}(H_{ext})\rangle$, where $\langle B^{down}(H_{ext})\rangle$ and $\langle B^{up}(H_{ext})\rangle$ are the calculated measured stray–fields during the down–and up–sweep of the magnetic field, respectively. The area under the ΔB_z curve is equal to the area of the hysteresis loop. In practice, with varying discrete measurement values for H_{ext}, it is necessary to interpolate before performing the subtraction. These background fitting methods are programmed into `ana.Hloop` (see Section 2.2.2).

3.2.3 Magnetic Fluctuations

Electric noise

The above–described background subtraction methods are changing the signal only marginally and, therefore, preserve vital information. This information is extracted through the resulting hysteresis loops and difference plots. The aforementioned high temporal resolution enables the analysis to additionally measure electric noise. This noise is amplified before it is processed by the lock–in, using a pre–amplifier, like the SR560 (Stanford Research Systems, see Fig. 3.3b, top left corner, teal). After both the pre–amplifier and lock–in amplify and filter the signal, it is relayed for post–processing.

SR785 and SR830DAQ

Two post–processing techniques are compared. First, the signal is relayed to the signal–analyzer SR785 (Stanford Research Systems, see Fig. 3.3b, bottom, orange). This instrument automatically applies the fast Fourier transformation (FFT) multiple times, averages it, and outputs the power spectral density (PSD) in the frequency domain. As a result of this process, the original time–signal is not accessible anymore. This absent signal is usually not burdensome when the emanating process is stationary, in place, and ergodic in time. In such a case, the signal follows the normal distribution, and the PSD represents the signal's entire noise. In contrast, the previous statement does not apply to dynamic processes, where a signal changes the mean value over time and loses its ergodicity. In aforesaid dynamic processes, higher–order power spectra (second spectrum) are necessary to determine the significant noise characteristics. Those noise characteristics use the temporal development of the time–resolved PSD (see Section 2.2.1). This information is accessible through the original time–signal measured by the lock–in's internal buffer. The corresponding EVE instrument function `SR830DAQ` (see Section 2.2.2, Page 12) is capable of performing automated routines, such as writing a list of commands that programs multiple instruments at once.

Magnetic noise

The combination of electric noise and magnetic characterization measurements opens exciting windows of opportunities. To seize these opportunities, I examined the magnetic noise under various conditions. The Hall bar's and structures' static noise is obtained by selecting and preparing stationary positions inside the hysteresis loop. The opportunity to measure long time–signals automated through EVE's routines allows the preparation of many equidistant positions inside the hysteresis. This approach generates an immense amount of data that is needed for the investigation of low–frequency dynamics. Additionally, various conditions during a changing external magnetic field are explored.

CHAPTER 4

RESULTS

This chapter investigates the nano–tetrapods' magnetic properties by exploring the response to an external magnetic field. The investigation explores symmetries and questions reproducibility to answer where and how magnetic noise emerges. First, the nano–tetrapods' ferromagnetic fingerprint is observed through hysteresis loops. The structures' comparison extends throughout selected angles θ of the external magnetic field H_{ext}. As expected, repetitions with identical initial conditions question reproducibility by yielding different fingerprints. To solve this puzzle, the measurements' noise is analyzed in the frequency space with the signal–analyzer. This analysis hints towards regions of interest in the hysteresis. Additionally, the data acquisition functionality of the SR830 allows an additional analysis of the temporal signal in these regions of interest. This additional analysis unveils noise in metastable states inside the hysteresis. Finally, the signal–analyzer's measurements are reproduced using the novel SR830DAQ functionality.

All displayed plots were created using `ana`'s test cases available via the supplemental information (see Appendix A). Interested readers are welcome to examine the source code and data. For this purpose, every caption includes the function name used to create the corresponding plot. Some legends and titles also include the measurement number, granting individual data exploration and employment of personal analysis techniques.

4.1 Magnetic Hysteresis Loops

Plusses (left) and Crosses (right) have a characteristic, angle–sensitive hysteresis loops (see Fig. 4.1 and 4.2). Plusses and Crosses display individual hysteresis loops at almost all angles. The average magnetic stray–field through the active area of the

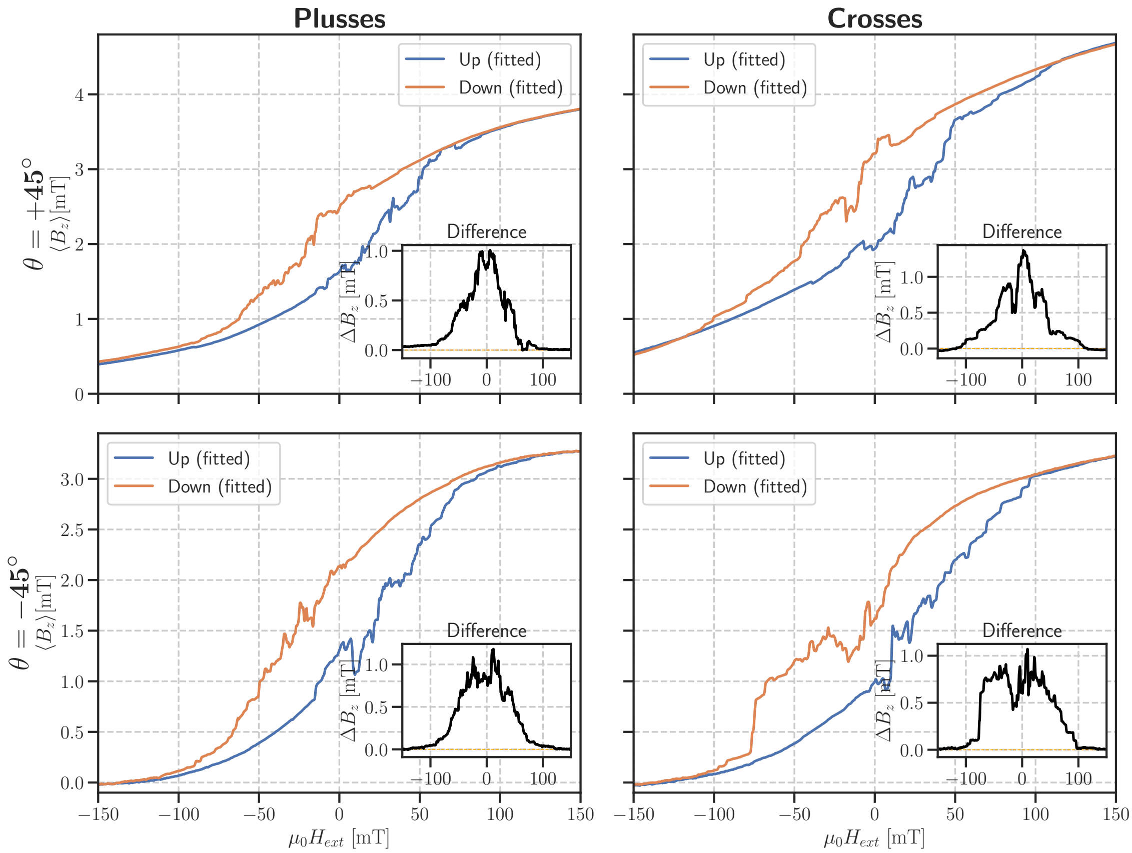


Figure 4.1.: **Hysteresis Loops ($\theta = \pm 45°$).** Magnetic hysteresis curves measured as magnetic stray–field $\langle B_z \rangle$ during up–and down–sweep of the external field H_{ext} at $T = 30\,\mathrm{K}$. The measured stray–field $\langle B_z \rangle$ is shown as a difference from the negative saturation. After setting the angle θ to $+45°$ (upper half) or $-45°$ (lower half), the Plusses (left) and Crosses (right) are measured subsequently using the gradiometry technique. Created by `test_plot_compare_hyst()`.

Hall sensor $\langle B_z \rangle$ is recorded during an up–and down–sweep of the external magnet. Therefore, the shown values for the magnetic stray–field $\langle B_z \rangle$ should be interpreted as relative values.

The Plusses and Crosses consist of multiple rotational and mirror symmetries (see Figure 3.1). Presumed inter–element dipolar coupling between single tetrapods may increment due to these symmetries. This dipolar coupling is expected to be stronger between the Plusses' tetrapods, as their arms link horizontally instead of diagonally, decreasing the interaction distance. Such magnetically coupled nanostructures may exhibit a complex switching behavior resulting in individually shaped hysteresis loops depending on various parameters.

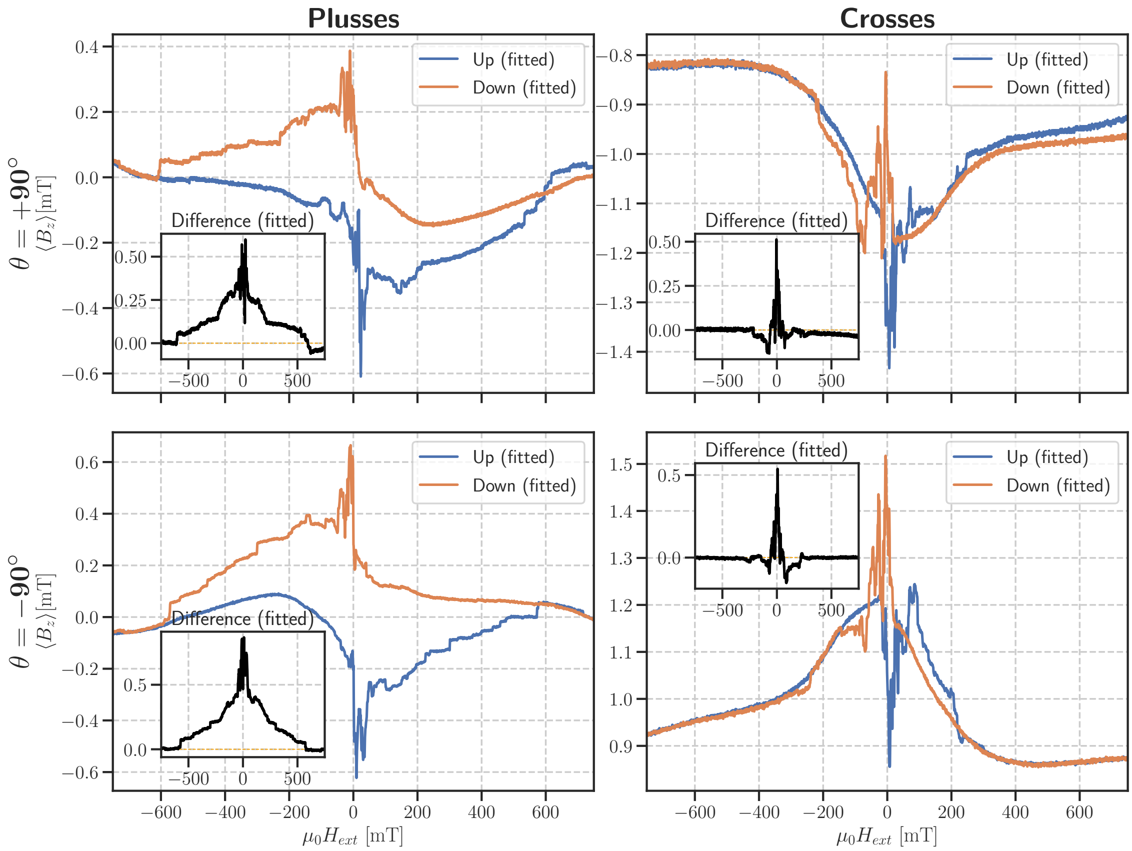


Figure 4.2.: **Hysteresis Loops ($\theta = \pm 90°$).** Magnetic hysteresis curves for an external field applied parallel to the sensor's surface. Measured Plusses and Crosses simultaneously with the parallel technique. Created by `test_plot_compare_hyst_par()`.

4.1.1 Angular Comparison

The hysteresis loops display a sensitive behavior concerning the external field's angular interaction on the nano–tetrapods' shape (Plusses or Crosses). First, at an angle of $\theta = \pm 45°$ (see Fig. 4.1), the external field acts nearly parallel to the x–z–plane of at least two arms (compared with Figure 3.2). The gradiometry technique applied here allows susceptible measurements, despite a large signal originating from the external field. These sensitivities offer several exciting insights.

All four hysteresis loops appear like smooth magnetization curves in most regions, except after passing the remanence on the path towards saturation. Within the first 50 mT after the externally forced magnetic polarisation's reversal, many spikes and steps occur in all hysteresis loops. The spikes develop very soon near the remanence. Later, between 50 to 100 mT, many separate small steps appear, indicating single

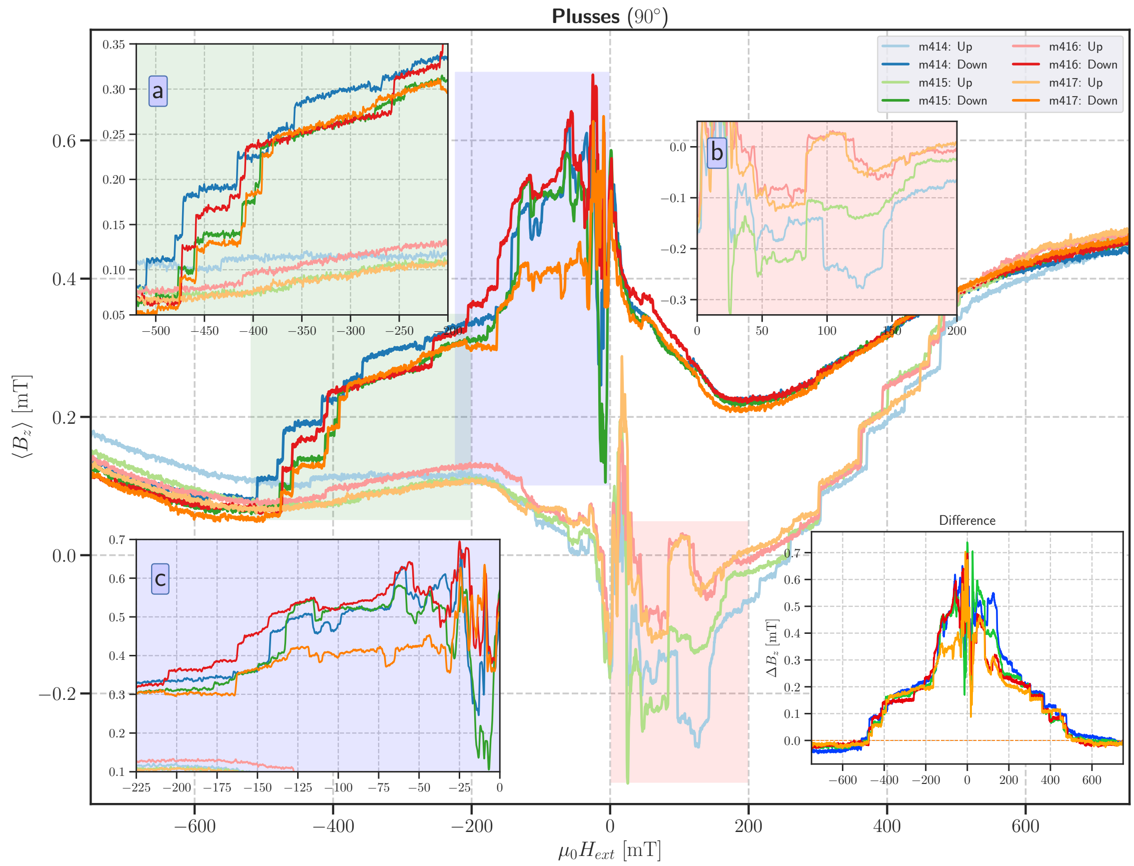


Figure 4.3.: **Repeated Hysteresis (Plusses).** Four subsequent repetitions of magnetic field sweeps in parallel configuration (90°) with identical initial experimental parameters. Up–and down–sweeps are highlighted in light and dark colors, respectively. Insets display a zoomed–in view of color–coded regions. The difference plot shows a significant variation between the measurements. Created by `test_plot_compare_multi_hyst_plusses()`.

switching processes. The spikes develop a stronger behavior at more extreme angles $\theta = \pm 90°$ (see Fig. 4.2).

However, for pair–wise particular angles, the loops are quite similar, for example, at $\pm 45°$ or $\pm 90°$. Notably, the particular angles of $\pm 90°$ open the window of opportunity to scrutinize the signal of the nano–tetrapods on top of the sensor. Generally, these susceptible signals are hidden beneath a superposed large background signal of the external field, as explained previously (see Section 3.2.2, Page 30). Nevertheless, by measuring the particular angles of $\pm 90°$, one can filter out the superposed signal without the gradiometry technique.

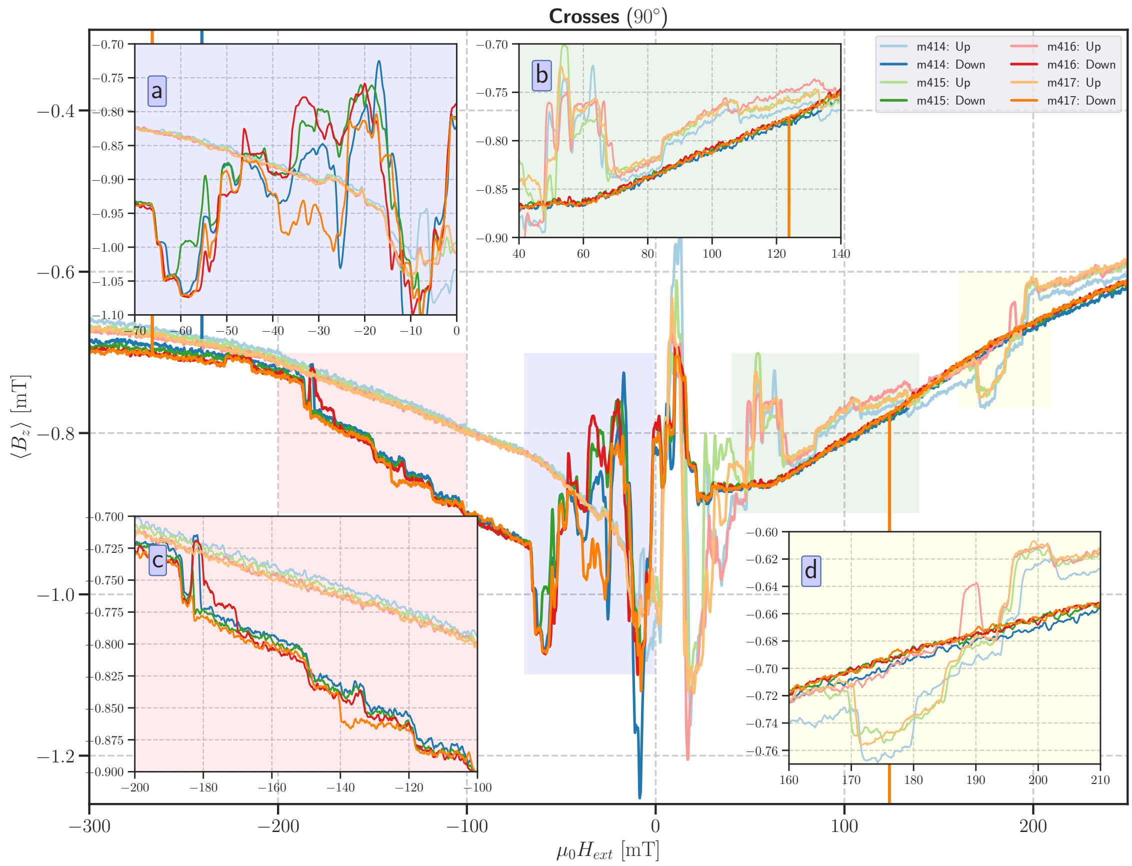


Figure 4.4.: **Repeated Hysteresis (Crosses)**. Same repetitions as in Figure 4.3. Created by `test_plot_compare_multi_hyst_crosses()`.

The Plusses' and Crosses' hysteresis loops develop unique characteristics at these particular angles of $\theta = \pm 90°$ without distinctly altering the general description above. A closer inspection of Figure 4.2 shows a remarkable similarity between plus and minus 90°. The contrasting curvatures indicate remnants of the background signal.

Micromagnetic simulations performed by Prof. Michael Huth (Goethe University Frankfurt) and macro–spin simulations executed by Prof. Christian Schröder (Bielefeld University of Applied Sciences) revealed details about the magnetization distribution at several states inside the hysteresis loop. The micromagnetic simulations suggest nucleation and propagation of vortex domain wall–like structures near the remanence [202]. These simulations further indicate that the hysteresis loop's shape is mainly influenced by the nano–tetrapods' anisotropy towards the external field (Plusses or Crosses). Those inter–element dipolar interactions mentioned earlier may affect the shape of the hysteresis loop in a subordinate role. Final simulations and discussions are still in progress.

To briefly summarize, the hysteresis loops are sensitive to the nano–tetrapods' orientation (Plusses or Crosses) and the external magnetic field's applied angle θ, except for pair–wise particular angles. Near the remanence, simulations suggest vortex–like magnetization states, represented by spikes in the hysteresis loop. Still, a relevant question remains. Are these results reproducible?

4.1.2 Repeated Hysteresis Loops

Figures 4.3 and 4.4 show subsequently measured hysteresis loops of the Plusses and Crosses, respectively. Zoomed–in color–coded insets expose a more detailed view of selected highlighted areas. These insets reveal a remarkable similarity between the measurements in regions without observable switching processes. As a side–note, Figure 4.4 contains some peaks with a seemingly approaching infinite gradient. These artefacts result from software bugs in EVE during the measurement, causing missing characters in the saved values. This problem only occurred once and could be solved. The data, however, were left untouched for integrity reasons and did not influence any results.

The general features of hysteresis are robustly reproducible (see Fig. 4.3 and 4.4). Nevertheless, a few details are not reproducible: the spikes that occur after passing the remanence on the path towards saturation (see Fig. 4.3b-c and Fig. 4.4a). When the external field drives domain walls through the nanostructures with characteristic pinning potentials, complex spin interactions occur. For such a quantum–mechanical process, the uncertainty principle may impose particular reproducibility restrictions. These restrictions only allow a stochastic description of the process.

The investigated nano–tetrapods' hysteresis loops are not uniquely reproducible. The magnetization curves of subsequently repeated hysteresis loops follow several nearly equivalent paths in the phase space that differ between cycles. This spread $\delta V_H = V_H - \langle V_H \rangle$ from the long–term mean $\langle V_H \rangle$ can be observed through a statistical ensemble of accumulated measurements. In the saturated (outer) areas of Figure 4.3 and 4.4, the repeated curves start and end in slightly different states. This behavior may result from thermal instabilities or capacitive effects in the experimental setup during the measurements, preventing the exact repetition of identical states. Nevertheless, the small ensemble of the four measurements shown here already identifies positions with considerable fluctuations. These fluctuations depend on the amplitude and the temporal gradient (up–or down–sweep) of the external magnetic field H_{ext}. This is the primary motivation to investigate these fluctuations for desirable, beneficial insights into dynamic processes.

4.2 Magnetic Flux Noise (MFN)

The noise (still defined as the spread δV_H from the mean) comprises several electric and magnetic components. The noise contributions originating from the detected magnetic flux are called magnetic flux noise (MFN). Unsurprisingly, there are more efficient ways to measure MFN than the one outlined before.

4.2.1 Signal–Analyzer (SR785)

While noise is typically examined under static conditions, our approach aims to investigate magnetization dynamics. The noise of such dynamic processes is more straightforward to measure by examining the magnetic response to an altering external field. Such an approach has been used by Bertotti to study magnetic Barkhausen noise [253–256]. His investigations exposed a $1/f^2$ behavior of the PSD [280, Ch. 9.3].

A state–of–the–art noise measuring technique is using a signal–analyzer. The signal–analyzer measures an electric signal and outputs the averaged Fourier transformation. The FFT over the entire sensor signal during a field sweep indicates the MFN's power–spectral density (PSD) in my experiment. The PSD is obtained during a field sweep and averaged over multiple bins. In this process, the signal's specific values are insignificant, as the fluctuations are filtered out.

Figure 4.5 shows the MFN's PSD for different field sweeps. For sweeps at large fields (blue and green lines), only frequency–independent thermal background noise is observable. However, when sweeping inside the hysteresis, the nanostructures' magnetic signal exclusively exhibits a $S_V \sim 1/f^2$ behavior. In contrast, the PSD of an empty Hall bar (light red) does not share this characteristic behavior. This is a remarkable observation, suggesting that the observed MFN indeed originates in the magnetization dynamics of the CoFe nano–tetrapods. The MFN spectra display no significant change considering the range of the sweep, as long as the field sweep covers the inside of the hysteresis.

Except for Figure 4.5b, all shown measurements up to here were measured at a temperature of 30 K. Figure 4.6 shows the thermal influence on the MFN's PSD. In this measurement, the nano–tetrapods are negatively saturated before measuring the signal during the up–sweep between $-25\,\mathrm{mT}$ and $+25\,\mathrm{mT}$. There is no significant difference in the noise in Figure 4.6. Therefore, all following measurements have been conducted at $T = 15\,\mathrm{K}$.

Nevertheless, the signal–analyzer can only provide highly aggregated results. It can output neither any time series nor allows it access to intermediate results. By now,

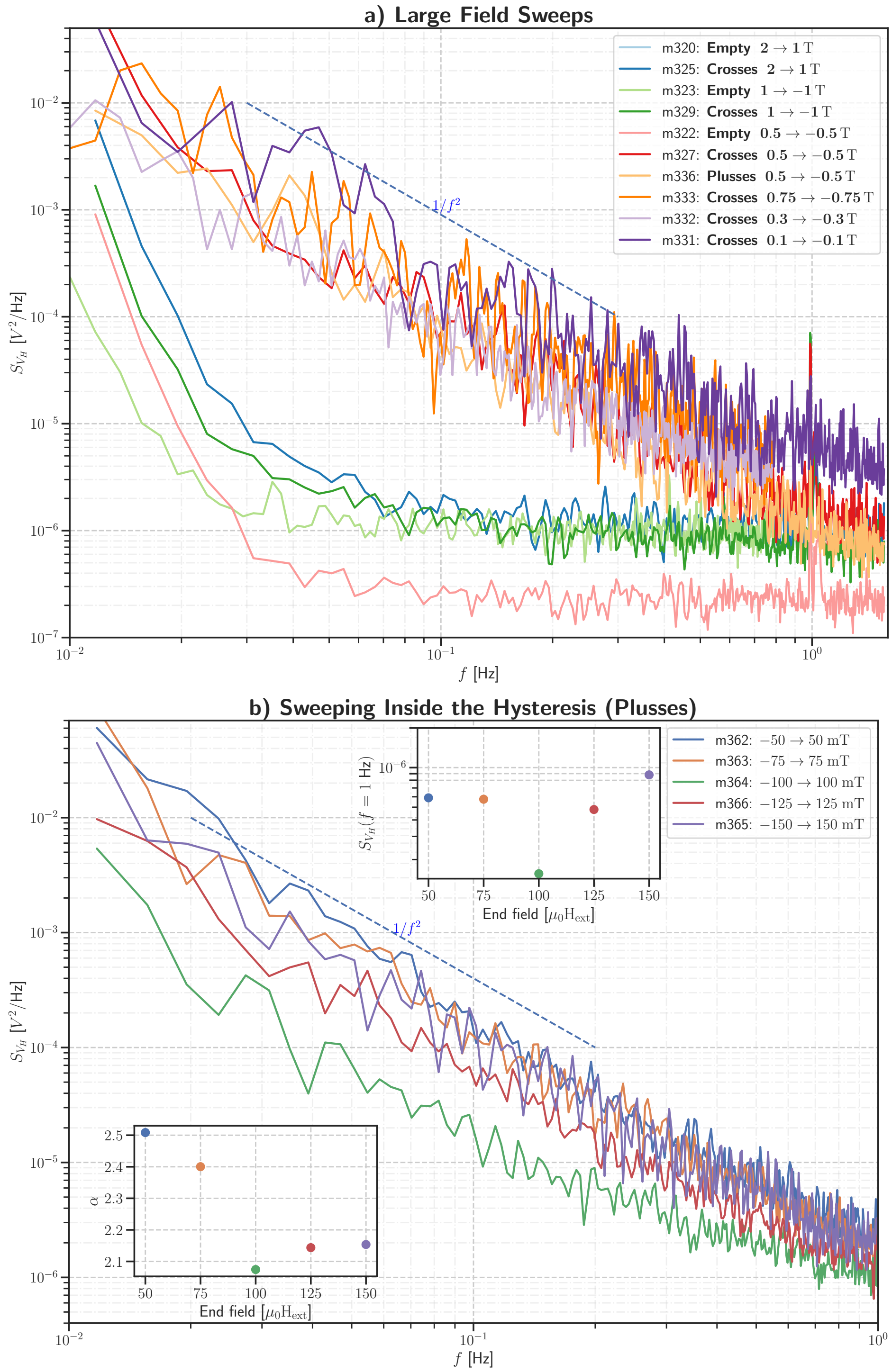


Figure 4.5.: **Noise PSD during field sweep.** Signal–analyzer (SR785) measurements of the MFN's PSD during an external field sweep. A dashed line represents $1/f^2$ behavior. Shown are the Hall signal's PSDs during **a)** field sweeps outside and covering the Hysteresis loop ($T = 30\,\mathrm{K}$), and **b)** multiple field sweeps of the Plusses after negative saturation with different start and end field positions ($T = 5\,\mathrm{K}$). Inset shows fitted noise amplitudes $S_{V_H}(f = 1\,\mathrm{Hz})$ and slope α from a linear regression between $20\,\mathrm{mHz} < f < 200\,\mathrm{mHz}$. Created by `test_sa_sweeps()`.

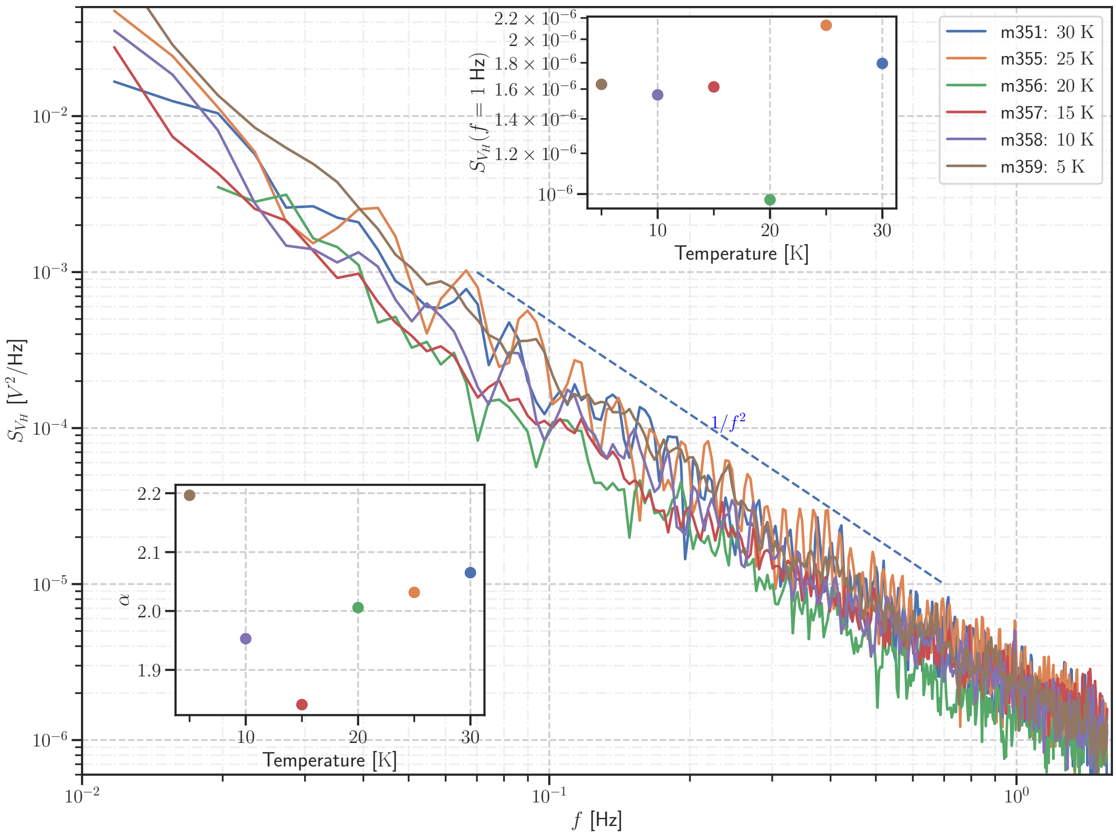


Figure 4.6.: **Comparison of different temperatures (Plusses)**. Multiple noise spectra of the Plusses' magnetic signal during the up–sweep between $-25\,\text{mT}$ and $+25\,\text{mT}$. Measurements were taken at various temperatures between $5\,\text{K}$ and $30\,\text{K}$ (see legend). Insets display resulting amplitude $S_V(f = 1\,\text{Hz})$ and slope α from a linear regression between $20\,\text{mHz} < f < 700\,\text{mHz}$. Created by `test_sa_temp()`.

we identified the position inside the hysteresis loop where magnetic noise occurs. In a noise–prone region of this two–dimensional phase space, the MFN's temporal development is scrutinized next.

4.2.2 Lock–In Data Acquisition (SR830DAQ)

In this experiment, the measurement is paused mid–way through the field sweep while the sensor keeps measuring to explore the MFN's temporal development. The stray–field's temporal fluctuations at static fields may be a more accurate indicator of the MFN, as they contain more extractable information than the one used in the previous section. This approach was inspired by Diao et al. [282].

The number of interruptions and measurements' durations defines how detailed the

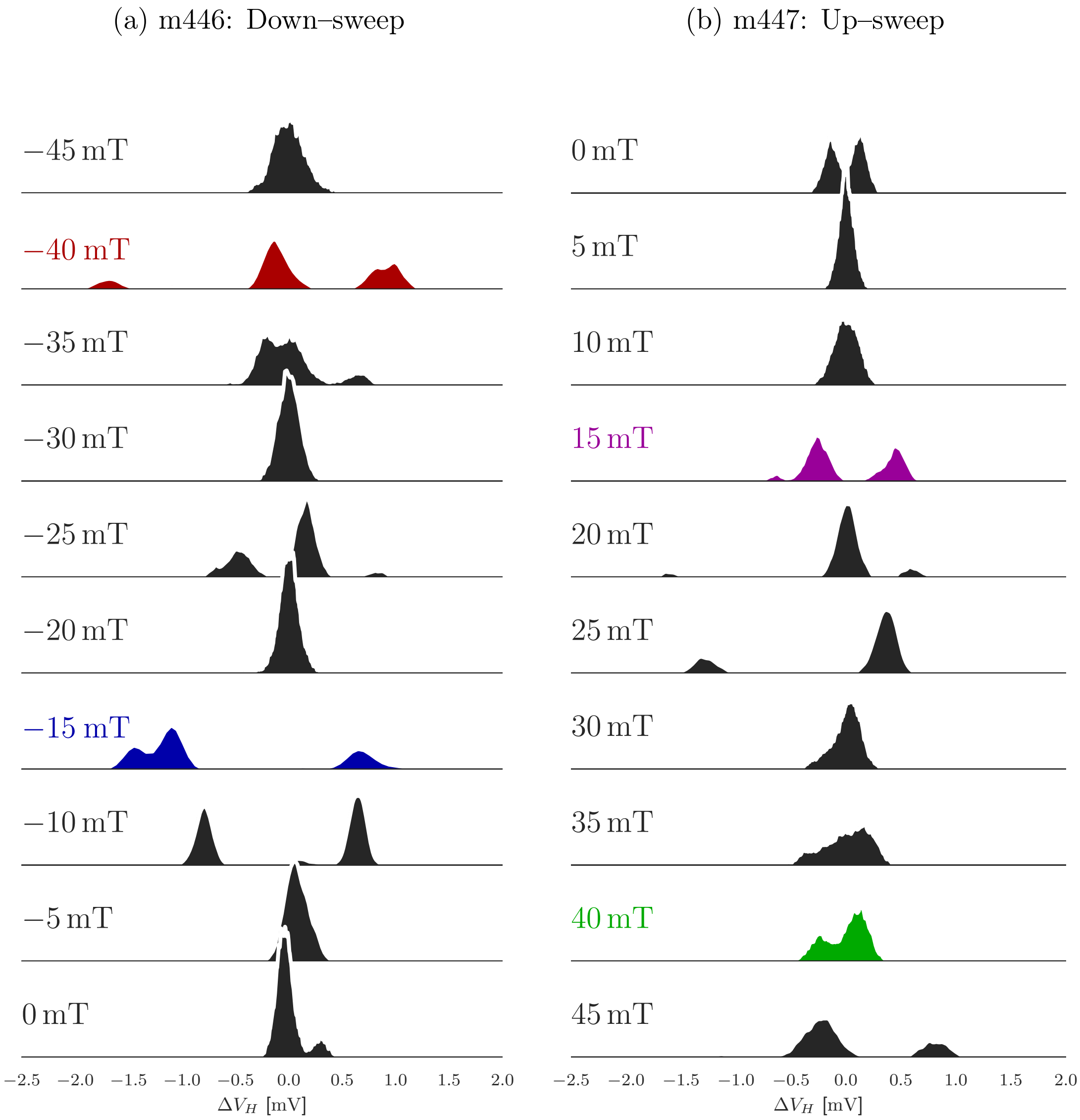
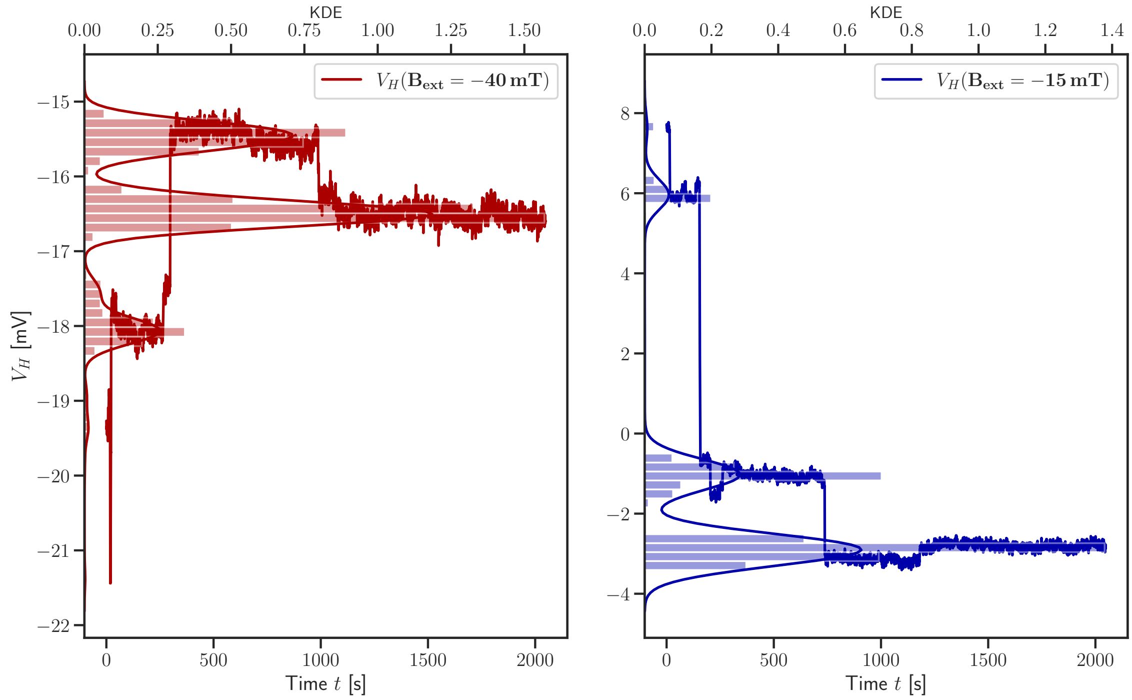


Figure 4.7.: **Time–signal's KDE at various field positions inside the hysteresis (Plusses).** Multiple subsequent measurements taken for 2048 s during an interrupted down–/up–sweep at specific field positions annotated left. The external field is applied in an angle $\theta = 45°$. Shown is the signal's KDE after subtraction of the mean value, representing the signal's distribution ΔV_H around the mean value. Created by `test_plot_hist()`.

(a) m446: Down–sweep



(b) m447: Up–sweep

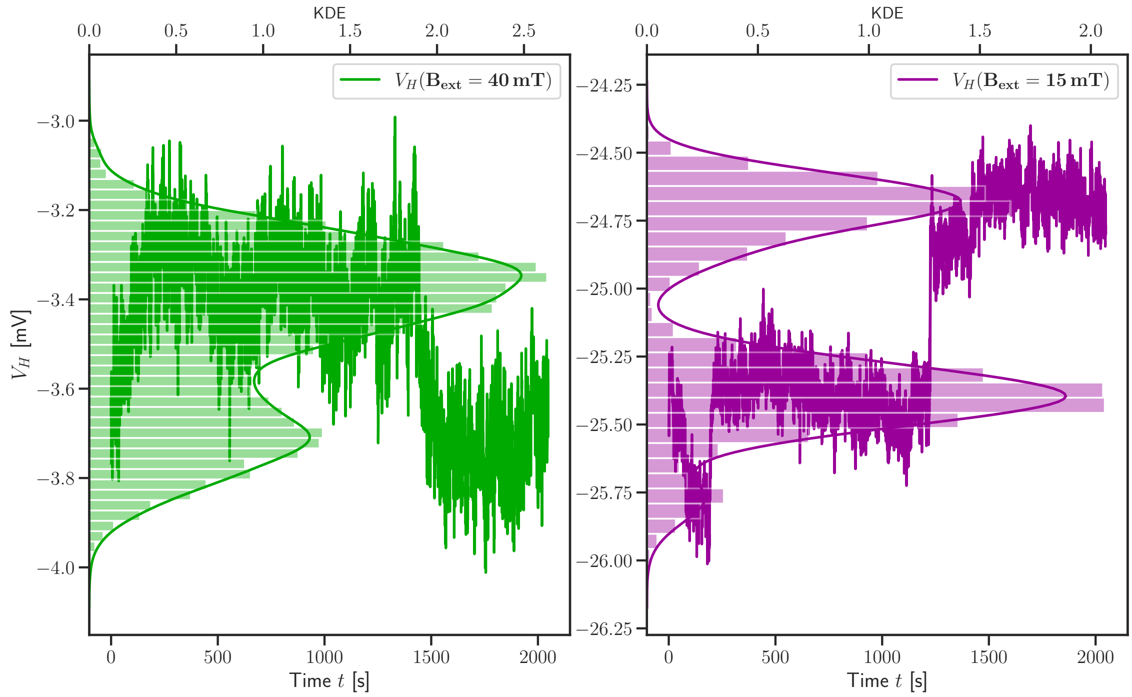


Figure 4.8.: **Time–signals at selected field positions inside the hysteresis (Plusses)**. Measured Hall voltage over time for 2048 s after stopping at specific field during down–/up–sweep, respectively. Solid lines show the time–signal in front (bottom axis) with a normalized histogram and it's outlining KDE in the background (top axis). The time–signals are color–coded to fit the colors of Figure 4.7. Created by `test_time()`.

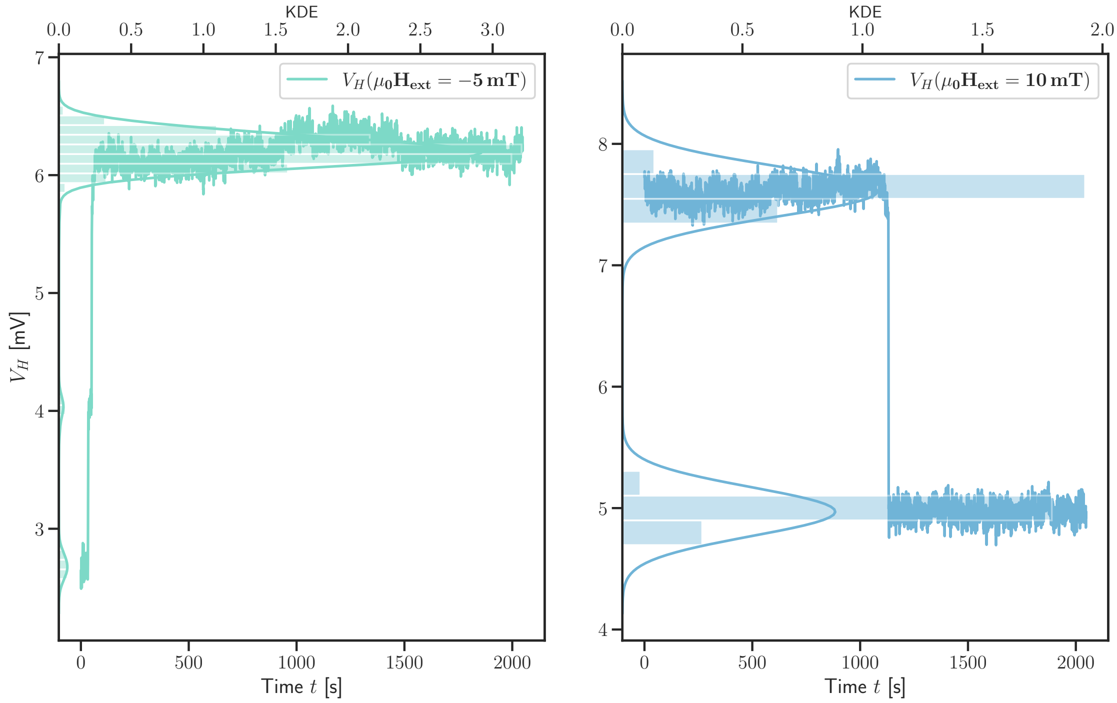


Figure 4.9.: **Time–signals at selected field positions inside the hysteresis (Plusses, m446).** Measured Hall voltage over time after interrupting a down–sweep. Solid lines show the time–signal in front (bottom axis) with a normalized histogram and it's outlining KDE in the background (top axis). The time–signals are color–coded to fit the colors of Figure 4.10. Created by `test_time2()`.

magnetic fluctuations inside the hysteresis loop are captured. Therefore, for the measurements in this section, the angle of the external magnetic field θ has been set to 45°. At this angle, the hysteresis loop is small enough to measure the inside using 40 interruptions between $\pm 100\,\text{mT}$ without changing the lock–in's sensitivity.

Time–signals

Figure 4.7 shows the kernel density estimation (KDE) of the detected signals at specific external field values noted aside. The KDE serves the time–signal's distribution ΔV_H around the mean and is normalized to represent a probability distribution. Shown here is only the noise–prone region right after passing the remanence. As a reminder, the spikes only occur in this noise–prone region of the hysteresis loops (as shown in Fig. 4.1 to 4.4).

There are a few fields where the signal has a single mean value and normal distribution. This is the expected behavior, which can also be observed in other hysteresis loop regions and empty Hall signals. Nevertheless, it is more common in the selected region that multiple mean values exist. This is the first evidence for a non–ergodic signal.

Figure 4.8a shows the raw measured signal at respectively highlighted positions in Figure 4.7. These time–signals reveal several distinguishable steps that indicate spin switching processes. This result is pivotal to explain the spread δV_H of the four experiments' mean in Figure 4.3 as it represents temporal instabilities at static fields — metastable magnetization states inside the hysteresis loop. Such spontaneous switching processes are known to appear in thermally activated systems [283]. Evidence for such thermally activated switching processes in similar CoFe nanostructures has been found by Mohanad Al Mamoori [202]. Consequential, it is reasonable to suggest that the spontaneous switching processes are thermally activated. In detail, thermal activation processes are statistically describe–able. Therefore, a bigger statistical ensemble of measurements could uncover further interesting statistical properties.

Analysis overview

The chosen method of measuring the time–signal yields large amounts of data. An analysis overview over each measurement gives the most informative details at a glance (see Fig. 4.10 and 4.11). The first plot (I.) displays the signal's PSD in the unaltered axis design from the last section. The second plot (II.) shows the PSD as a contour plot with the different external field positions on the x–axis and the frequency on the y–axis. As in the previous visualization, frequency and PSD axes are displayed logarithmically.

In the third plot (III.) of each analysis overview, a linear regression applied at frequencies below 2 mHz determines the PSD's slope α. This slope is significantly elevated in Figure 4.10 at the highlighted field positions of Figure 4.8a. Two extra signals inside the noise–prone region disclose slopes larger than 1.5. are at external fields -5 mT and 10 mT. Closer inspection of the time–signal at those fields (see Fig. 4.9) reveals a single confined step with a magnitude of multiple mV. Other time–signals at larger fields do not exhibit such steps.

Comparable to the slope, the integral (IV.) expresses the power of the noise. This integral shows a relationship like $S_V(B) \sim B^2$. This result reproduces expectations from previous findings [211, 284] and indicates the Hall device's background noise. The presented time–signals from Figure 4.8a and 4.9 (down–sweep) feature an elevated noise power well above the background noise (see Fig. 4.10). Unfortunately, this could not be reproduced with the noise powers or slopes in Figure 4.11 (up–sweep).

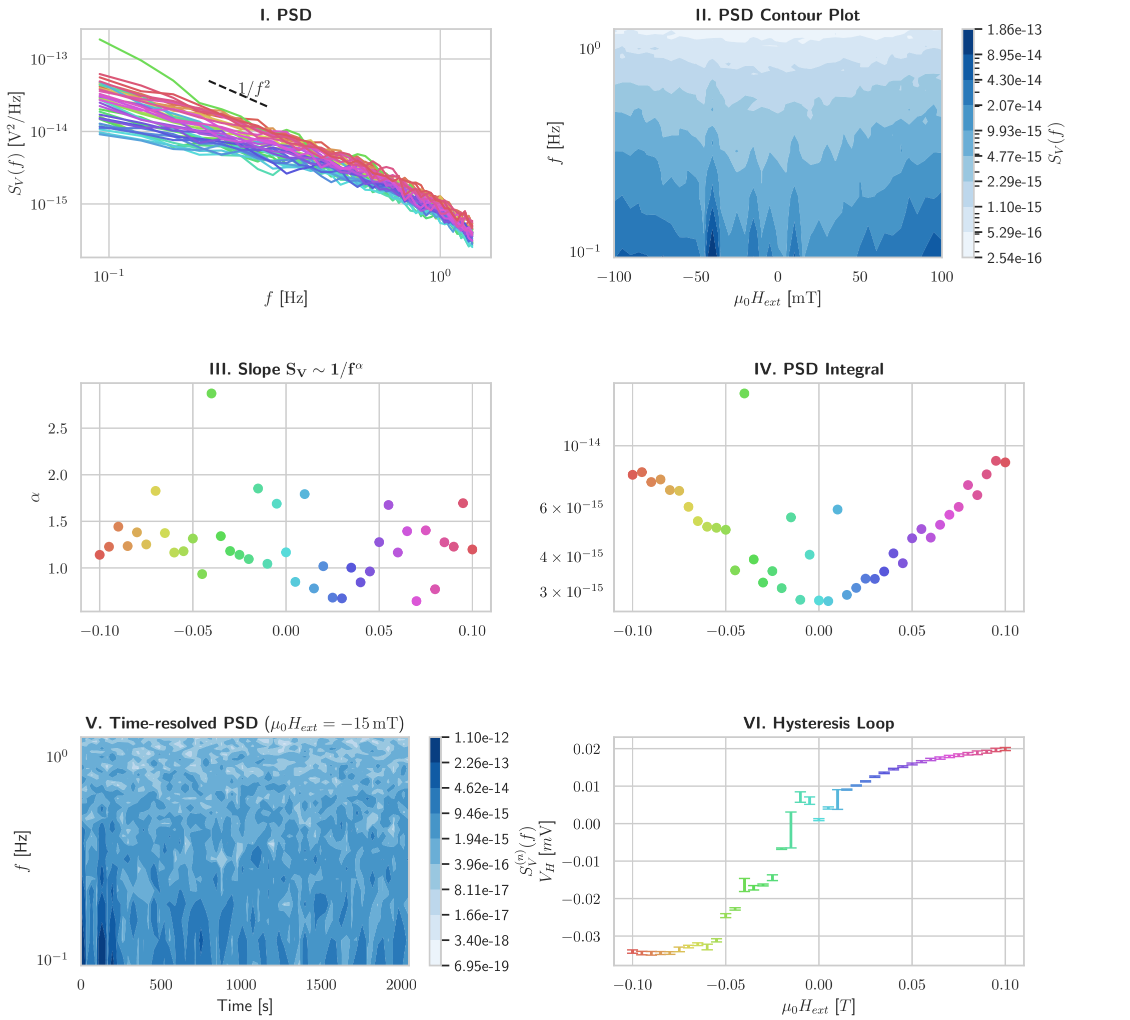


Figure 4.10.: **Analysis overview: Down–sweep (m446)**. Multiple plots show extracted information from the analysis of MFN measurements during an interrupted field sweep. See text for details. Created by `test_info()`.

The fifth inset plot (V.) shows the time–resolved PSD at a single position inside the hysteresis. This time–resolved PSD is automatically created through the **`spectrumanalyzer`**'s algorithms (see Section 2.2.1). Similar to the PSD's contour plot (II.), the y–axis displays the frequency, and the z–axis displays the PSD $S_V^{(n)}(f)$, but the x–axis represents the temporal dimension. The time–resolved PSD in Figure 4.10 shows several identifiable spikes (dark blue) at times $t < 400\,\mathrm{s}$, which, compared with Figure 4.8a (right, blue time–signal), determine temporal areas with considerable magnetization jumps. Furthermore, around $t \approx 750\,\mathrm{s}$, a separate, smaller spike pinpoints the last significant switching process discoverable in Figure 4.8a.

The last inset plot (VI.) displays the captured signal as an error bar over the external field position. The error bar's size is twice the variance of the signal ($\Delta V_H = 2\sigma$).

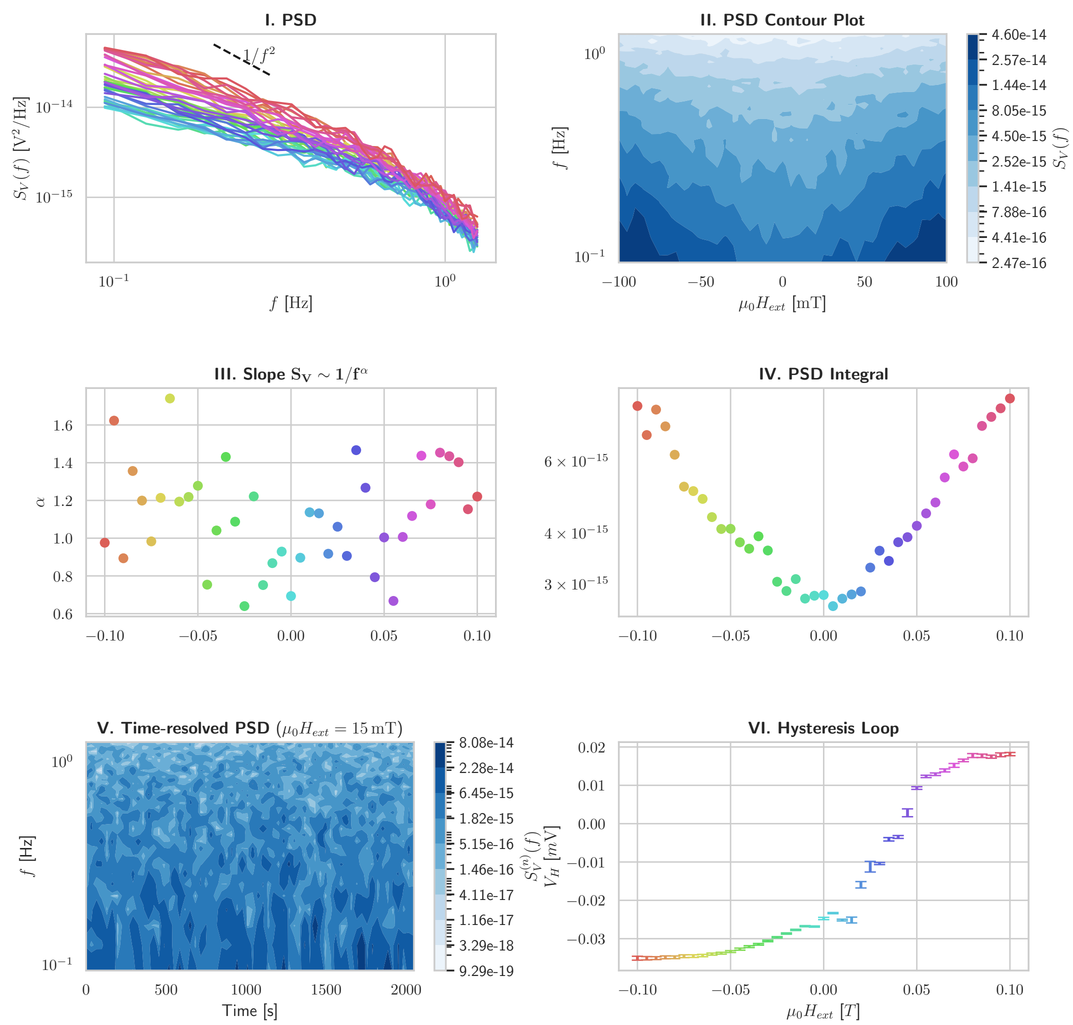


Figure 4.11.: **Analysis overview: Up–sweep (m447)**. Multiple plots show extracted information from the analysis of MFN measurements during an interrupted field sweep. See text for details. Created by `test_info()`.

4.2.3 Method Comparison

The results of both methods (SR785 and SR830DAQ) are compared in equivalent situations to ensure the comparability of both measurement techniques. The experiments in this section follow the same approach as related measurements with the signal–analyzer SR785 (presented in Section 4.2.1).

All measurements in this section investigate the Hall signal's PSD during the up–sweep between $-25\,\mathrm{mT}$ and $+25\,\mathrm{mT}$ after negative magnetic saturation. The external field is applied at an angle of $\theta = 90°$, and the parallel measurement configuration is employed.

The experiments were performed with small variations: The signal–analyzer (SR785) produces better results with a pre–amplifier. This pre–amplifier magnifies the detected signal by a factor of 20 but also augments its noise. However, the SR830DAQ measures the noisy signal directly without an intermediate pre–amplifier. The signal–analyzer also measured with a different time–constant setting in the lock–in to acquire more extensive frequency ranges. A time–constant of $\tau = 3\,\mathrm{ms}$ (SR785) averages faster, leaving more noise residues in the signal. The SR830DAQ obtained the values at a time–constant of $\tau = 100\,\mathrm{ms}$, as a sampling rate of 8 Hz has proven to produce the best results. These variations may lead to a scale difference of multiple magnitudes in the PSD.

Sweeprates

First, both methods compare various sweeprates, the velocity of the external magnetization's change. Figure 4.12a shows the resulting PSD output of the signal–analyzer (SR785). Except for the slowest sweeprate, all other sweeprates unveil a noteworthy similarity. Figure 4.12b shows the analysis of the SR830DAQ method. Both methods exhibit a clear $S_V \sim 1/f^2$ behavior for all sweeping measurements, independent of the sweeprate.

A closer inspection of the insets, specifically the calculated amplitude $S_{V_H}(f = 1\,\mathrm{Hz})$, exposes a correlation between the sweeprate and the PSD's amplitude beneath $d/dt(\mu_0 H_{ext}) \leq 1\,\mathrm{mT/min}$. This correlation is hardly detectable and only becomes visible without the calculated fit, when focusing on sweeprates $d/dt\mu_0 H_{ext} \leq 0.25\,\mathrm{mT/min}$. It is reasonable to conclude that a lower sweeprate is accompanied by a clear decrease in the noise power, at least for very low sweeprates. However, considering the inset's scale, for sweeprates $d/dt(\mu_0 H_{ext}) \geq 0.5\,\mathrm{mT/min}$ this statement should be considered uncertain.

Unfortunately, a software bug prevented posterior repetition of such measurements at low sweeprates to determine the this behavior's tenability. In the attempt to repeat these low sweeprates' measurements, the EVE instrument module for the magnetic power supply `IPS120` could not start the sweep of the magnet. Consequentially, the black line displays a single measurement without any field change (at $\mu_0 H_{ext} = -25\,\mathrm{mT}$ with preceding negative saturation), as opposed to the other's $1/f^2$ behavior.

Current amplitudes

Figure 4.13a shows the same $\pm 25\,\mathrm{mT}$ sweep–range for various driving current amplitudes. It is expected that a current dependence would scale the noise like $S_V \sim I^2$. This expectation could not be verified with the signal–analyzer (SR785). Instead,

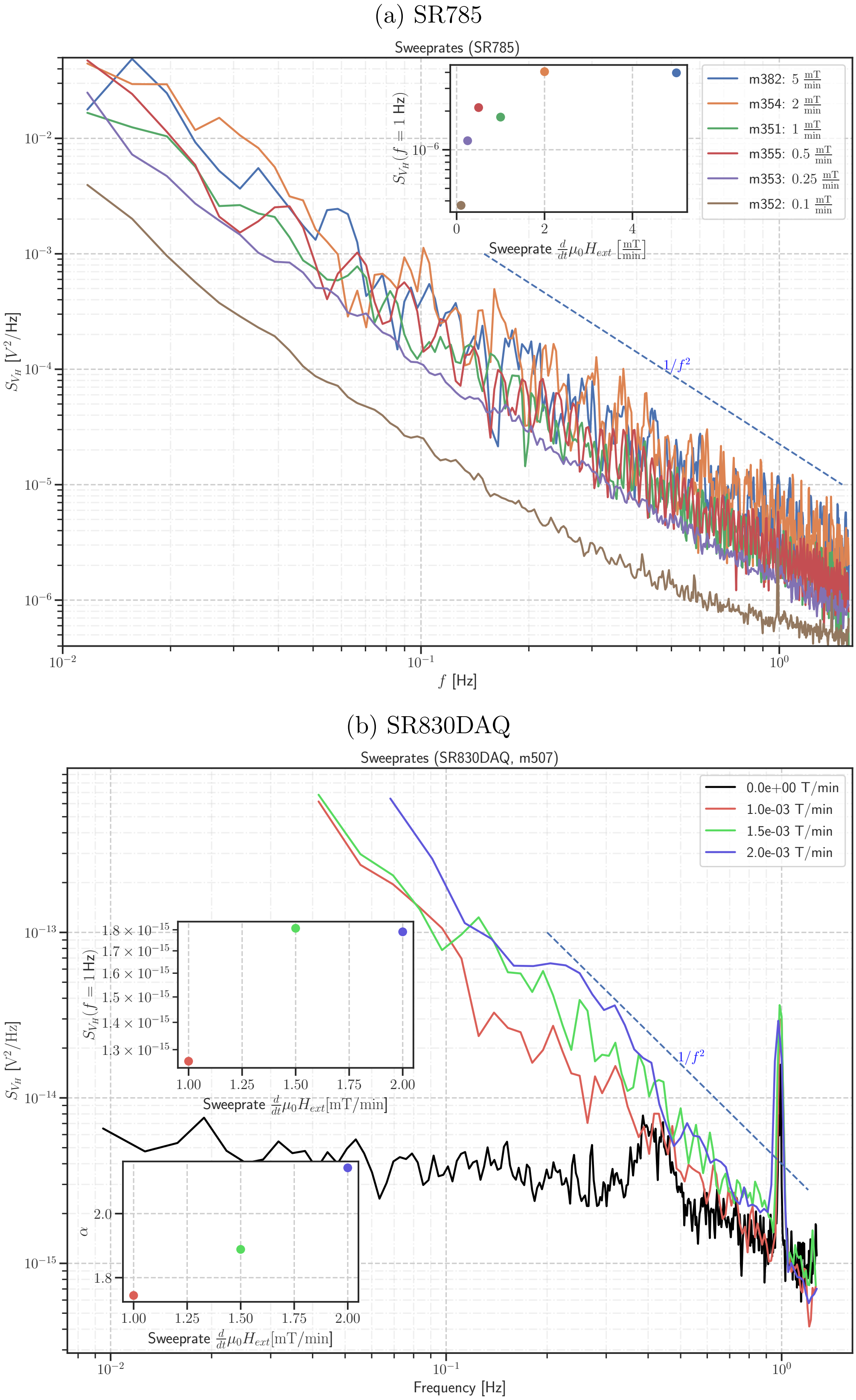
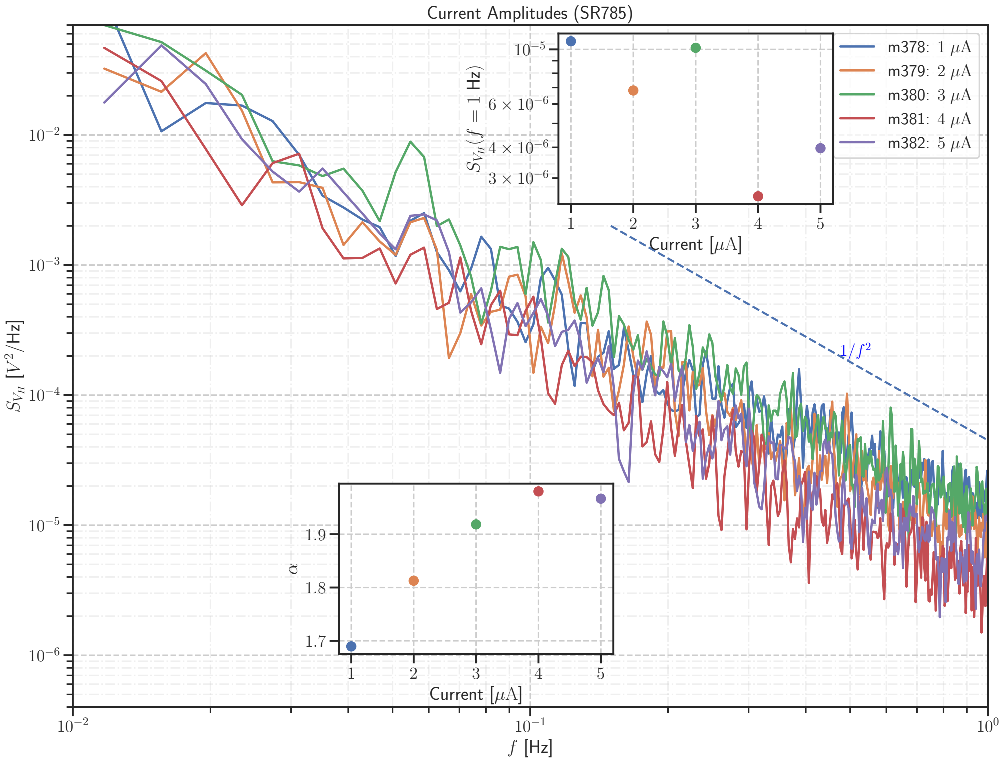


Figure 4.12.: **Comparison of different sweeprates**. Repeated measurements of similar conditions to compare both measurement techniques. See text for details. Created by `test_compare_sweeprates()`.

(a) SR785



(b) SR830DAQ

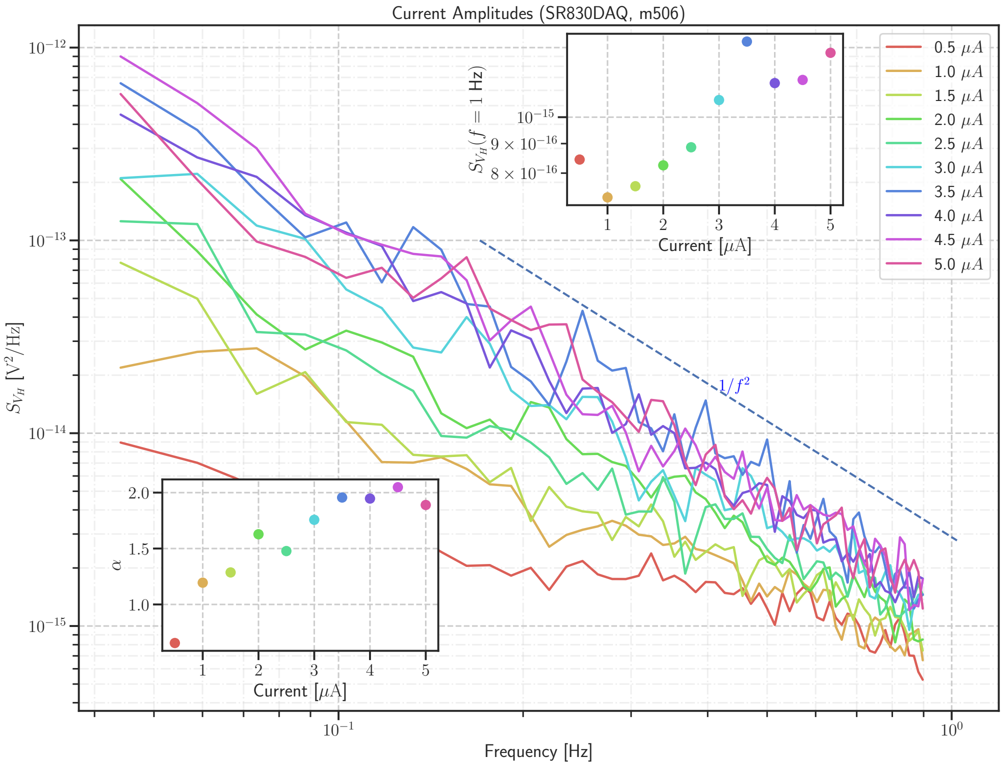


Figure 4.13.: **Comparison of different current amplitudes**. Repeated measurements of similar conditions to compare both measurement techniques. See text for details. Created by `test_compare_current()`.

the PSD's amplitudes are remarkably insensitive to changes in the applied current. Although the inset's result of the fitted PSD amplitude exposes a detectable correlation between the current and PSD, this correlation is doubtful when considering the PSD's scale and omitted error bars.

Posterior repetition of the experiment with the SR830DAQ (see Fig 4.13) yields the expected behavior. The fitted amplitude $S_V(f = 1\,\mathrm{Hz})$ displays a clear $S_V \sim I^2$ relationship, which also becomes evident in the PSD.

Interestingly, the slope α also displays a correlation with the current I, noticeable with both methods. This correlation between α and I is remarkable because it has not been discovered in the literature yet and is worth further investigation.

To summarize, the measured hysteresis loops of the nano–tetrapods are sensitive to the relative orientation of the tetrapods towards the external field. Subsequent repetition of identical experimental parameters yields several nearly equivalent hysteresis loops with identifiable noise–prone regions. In such noise–prone regions, the magnetic stray–field at static states exhibits spontaneous switching processes observable via distinguishable steps in the time–signal. The analysis framework `ana` provides an analysis overview that grants insight into the most significant statistical details of multiple measurements at once. This analysis overview helps to quickly identify individual time–signals with spontaneous switching processes. These switching processes are deduced to indicate metastable magnetization states inside the hysteresis, which may originate from thermally activation. The frequency–dependent analysis of the MFN scrutinizes the signal's PSD with a signal–analyzer during a field sweep and displays a clear $S_V \sim 1/f^2$ behavior, which only occurs when sweeping over noise–prone regions. Noteworthy, this behavior is insensitive to changes of the temperature or external field's sweeprate. A comparison between both methods in similar conditions displays a good reproduction of this $1/f^2$ behavior with the new SR830DAQ method.

CHAPTER 5

DISCUSSION

The initial objective of this study was to identify and characterize magnetic fluctuations. As a test system, we chose three-dimensional ferromagnetic CoFe nano–tetrapods. These nano–tetrapods were grown via FEBID on top of a home-built micro-Hall sensor, a collaboration project with Prof. Michael Huth (Goethe University Frankfurt). In order to discover the nano–tetrapods' specific properties, the magnetic response to an altering external field is scrutinized. Two statistical analysis methods, precisely a signal–analyzer SR785 and a novel data acquisition technique (SR830DAQ), determine the signal's PSD to observe frequency–dependent noise.

The SR830DAQ technique utilizes previously unused functionalities of the lock–in amplifier SR830 to dissect the lock–in's time–signal directly. The presented results serve as a »proof of principle« demonstration of this developed measurement technique. It allows high–resolution measurements of the Hall signal's temporal development. This temporal investigation reveals metastable magnetization states in noise–prone regions, which are directly observable in the acquired time–signal. Additionally, repetition of similar measurement conditions corroborates findings of $1/f^2$ noise when sweeping the field over noise–prone regions near the remanence. Follow–up experiments will validate the equivalence of both methods on structures with well–known behavior.

The Hall signal's PSD can indicate noise characteristics presumably emanating from these metastable magnetization states. The introduced Python data analysis framework `ana` provides an analysis overview for such interrupted MFN measurements, which can help identify such metastable magnetization states. This analysis overview revealed an elevated slope and power in the noise of time–signals that contain observable steps. Sizeable steps can be additionally identified in the time–resolved PSD. In other artificial spin–ice systems, thermally induced magnetization switching processes have been observed [283]. This investigation includes a statistical »survival time« analysis. Such an analysis could further scrutinize the metastable magnetization states' statistical characteristics and corroborate the deduction that such metastable magnetization states originate from thermal activation processes.

Furthermore, the Hall signal averaged over a field sweep inside a noise–prone region always yields a characteristic $1/f^2$ spectrum, possibly indicating domain wall motions [280, Ch. 9.3]. These characteristic spectra are remarkably insensitive to changes in the temperature or external field's sweeprate. Unlike Bertotti [280, Fig. 9.14], the noise presented in this study appears exclusively in very low frequencies and does not exhibit a sweeprate dependence.

The correlation between the PSD and current amplitude could only be observed with the SR830DAQ method. Measurements with the signal–analyzer (SR785) of a pre–amplified signal could not reveal such a correlation. These contrasting results deserve further investigation. Because of the metastable magnetization states in the observed region, the PSD of such measurements is also more meaningful for larger statistical ensembles than those presented. Such larger ensembles would further allow the determination of accurate error bars for the applied linear regressions. However, `ana` does not yet analyze the error of calculated fits. The insets' scales of presented PSD's regression results suggest that the error explains single outlier. The presented fitting results and their errors highly depend on the range where the fit is applied. Except where noted otherwise, the default fitting–range is between $20\,\mathrm{mHz} < f < 700\,\mathrm{mHz}$. These values have proven useful to avoid the inclusion of artefacts.

The angle of the external magnetic field θ is determined through the Hall signal's proportion to the maximum signal at $\theta = 0°$. Because of thermal instabilities and their influence on the Hall effect, this proportion can not be determined with exact confidence. Therefore, every change of the angle θ results in an error of approximately $\Delta\theta \approx \pm 2 - 5°$. Subsequent measurements are always performed at identical angles, but exact posterior reproduction of specific measurements at precise angles is only possible to a limited extend.

The magnetic stray–field's calculation from the Hall voltage assumes an ideal squared active area of the Hall cross and a constant applied current of $I = 2.5\,\mu\mathrm{A}$. Diffraction effects in the fabrication process result in rounded edges (see Fig. 3.1). The calculated magnetic stray–field should take a Hall response function $F_H(x, y)$ into account to consider ballistic and diffusive transport regimes [285, 286]. However, the data analysis in this thesis assumes the response function to be a constant $F_H(x, y) = 1$, as the estimation demands extensive calculations and exceeds this thesis's scope. This negligence results in a relative error, which is presumed to be approximately 30 %. The effect of the applied current on the stray–field's calculation is discussed next.

The applied current varies only slightly when applying the gradiometry technique. In this configuration, the limiting resistors R_1 and R_2 manipulate both applied currents I_1 and I_2, respectively. These limiting resistors were usually adjusted between $1\,\mathrm{M\Omega} \geq R_{1/2} > 1.05\,\mathrm{M\Omega}$ in order to balance the gradiometry technique for individual angles. An applied voltage of $V_{in} = 2.5\,\mathrm{V}$ results in an effectively applied current of $I_{1/2} \approx 2.5\,\mu\mathrm{A} - \delta I_{1/2}$ where the difference $\delta I_{1/2}$ to the assumed applied current has an upper boundary of

$$\delta I_{1/2} \leq \frac{2.5\,\mathrm{V}}{1\,\mathrm{M\Omega}} - \frac{2.5\,\mathrm{V}}{1.05\,\mathrm{M\Omega}} \approx 0.12\,\mu\mathrm{A}.$$

The propagation of this neglected current difference $\delta I_{1/2}$ into the calculated stray–field is marginal.

The present study was designed to comply with current best practices for reproducible and reusable research, summarized in the introduction. All acquired data, documentation, and algorithms are available via the supplemental information (see Appendix A). The supplemental information was initially composed to support the experimentalist's future–self during the experiment. After finishing the experiments, a reasonable amount of time is invested in adapting the supplemental information to a presentable web–page for reviewers and succeeding scientists. External contributions receive credit, even without legal obligation (see Appendix B). Except for EVE's and `spectrumanalyzer`'s source–code, every created code, figure, and text is licensed (see Appendix B) to optimize re–use potential. `ana` includes automated test cases using GitLab CI.

The measurements are solely analyzed with `ana`, making reproduction and re–use easier. `ana` was initially created to analyze and visualize the data acquired in the course of this study. The algorithms used to read and process the data are customized for exercised filename and file–structure conventions. The source–code still contains several equivalent passages that could be extracted into separate functions. Generalization of single customized passages could establish a comprehensive analysis framework.

The introduction of a GitLab server improved software development, source–code control, and documentation management. The GitLab server expedites code maintenance and intensifies collaboration if accepted by users. User acceptance is a critical success criterion, as the benefits mentioned above only arise when users welcome the new environment. Therefore, user acceptance determines GitLab's success and impact on future research.

In recent years, software products depend more than ever on open–source components. All open–source projects contain vulnerabilities that need years to be found, fixed, and updated [287]. Proprietary software with closed–source concepts also contains vulnerabilities that are not communicated and, therefore, could radically exacerbate software security even further. To fast–track the finding process of such vulnerabilities, GitLab has launched a bug bounty program [288], motivating the community to delve into security tests searching for critical bugs. Such a bug bounty program awards the finder of security issues with monetary appreciation, facilitating faster security patches.

It is well known that all Docker containers inherit these above–mentioned security flaws [289]. This problem is universal and unavoidable, regularly revealing severe exposures [290]. State–of–the–art solutions to this issue include the setup of monthly updates, monitoring systems [291], and security scorecards [292]. The decision to provide supplemental information online consequentially spotlights a risk–assessment of the employed server. The risk of individual vulnerabilities is commonly assessed by estimating its severity and occurrence probability [293]. For the employed server, minimized risks are presumed. In detail, high–entropy password protection of critical systems and presumed scarcity of interest diminish potential risks. This and other possible improvements are highlighted next.

5.1 Outlook

As security flaws and crashed systems are always an issue, preparation for data loss is imperative. The configured GitLab server includes a built–in backup functionality. This backup is configured for automated weekly backups. This automated backup is only stored on a local hard drive. For future endeavours, it is highly recommended to automatically store at least one recent backup offline and off–site.

The commitment for clean code and precise tests can be advertised with a »CII Best Practice Badge« [294]. Such a badge has critical requirements on code quality, test coverage, and other project internal obligations. All those at first glance seemingly negative liabilities can amplify collaboration efficiency and proliferate popularity — effectively boosting the audience.

To help fulfill the above–contoured obligations, the open–source community provides excellent tools for improved automated testing [295] and code analysis [274]. Recently, GitLab published free CI security templates with optional premium features. These premium features are continually integrated into the free version [296]. The templates include license and vulnerability scanners[1]. Listing 2.1 shows how to apply the templates, and Figure 5.1 shows the output of a premium vulnerability scan on the Jupyterlab Docker image.

[1]Result from license scanner for the `jupyterlab` Docker image is openly available: `https://gitlab.com/ganymede/jupyterlab/-/licenses`

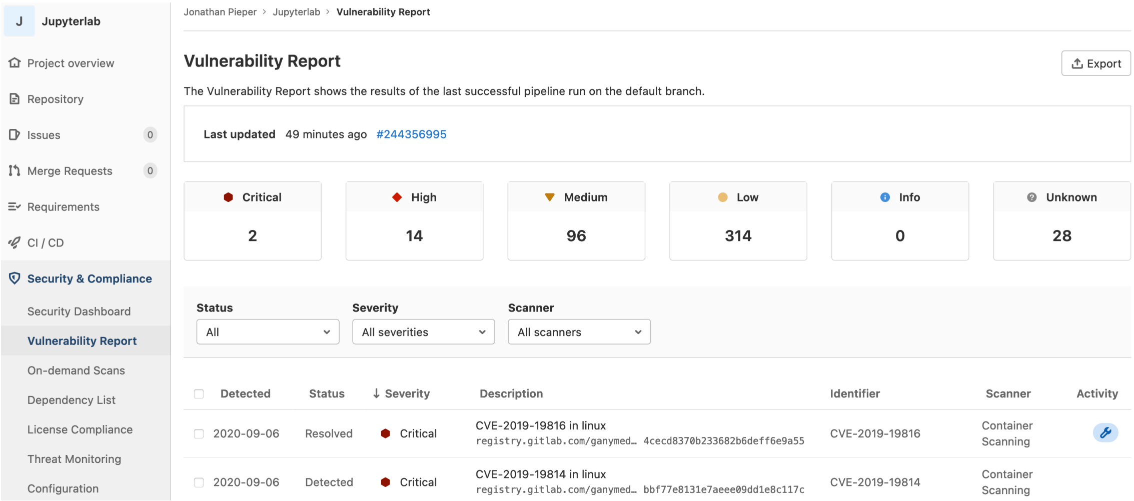


Figure 5.1.: **GitLab Vulnerability Report**. GitLab premium features allow to scan for vulnerabilities in Dockerfiles.

Python provides various packages and APIs to improve visualizations for enhanced data representation [297]. Some examples demonstrate how data can be differently presented nowadays [298–300]. Several open–source tools can connect a background Python kernel, interacting with the user while displaying the visualization [301]. Utilizing such tools allows the presentation of an interactive visualization on a web–page. Additionally, open–source tools can help scientists exploit further communication tools to expand narrative possibilities and advance collaboration [302].

If the GitLab server can be effectively deployed and benefits a growing number of students, collaboration could expand even further through adaptation into a university–wide GitLab server with premium functionality for all students. GitLab provides special educational licenses for such purposes [303] used by various universities [304–311].

A large amount of static data is not supposed to be version controlled using git. Unfortunately, in the restricted time–frame, no other tool seemed suitable. For future endeavours, I would recommend a data version control system [312].

In condensed–matter physics, also the sample fabrication improves from advanced data science. Novel machine learning approaches can optimize fabrication techniques of electromagnetic nanostructures and nanoelectronic devices [313–316].

The presented data acquisition method enables new possibilities to further scrutinize the time–signal. In detail, there are numerous classical analysis tools that already provide programming interfaces for effortless usage [317, 318]. Additionally, novel time–series analysis methods utilize modern machine learning techniques and statistical models [319–324]. Machine learning algorithms could apply linear regression models $y(\vec{x}, \vec{w}) = w_0 + w_1\phi_1(x_1) + \cdots + w_N\phi_N(x_N)$ with various basis functions ϕ to learn the ideal weights $\vec{w}$ for a given time–signal [325]. Such basis functions could consist of multiple trigonometric functions to mimic a Fourier transformation. Additionally, extra basis functions, like the sigmoid function $\phi_n(x) = (1 + e^{-x})^{-1}$, could help identify steps in the time–signal.

CHAPTER 6

SUMMARY AND CONCLUSION

The present study aims to find and characterize fluctuations in the magnetic fingerprint of three–dimensional nano–tetrapods. These tetrapods were deposited by means of focused electron beam induced deposition (FEBID) on top of a micro–Hall magnetometer. The micro–Hall measurements scrutinized the magnetic fingerprint by measuring the nanostructures' stray–field during an external field sweep and obtaining the hysteresis loop. Repetitions of identical experiments yield several, nearly equivalent hysteresis loops that differ in noise–prone regions near the remanence. These noise–prone regions are further investigated using statistical methods to determine the noise's power spectral density (PSD).

During the process of data acquisition and analysis, utmost efforts were employed to comply with current best research practices. In general, the data processing steps involve customized, self–written computer algorithms that are freely available and fundamentally documented. Based on experiences, the workflow's documentation process evolved from proprietary OneNote notebooks towards an open–source driven Continuous Analysis infrastructure, allowing automated data analysis and interoperable documentation. In detail, a self–maintained GitLab server provides the infrastructure to manage git repositories that version–control data, code, and documentation. In addition, Continuous Integration tools automate tasks and increase productivity. These aforementioned data processing and workflow steps have been fundamentally documented, converted into a presentable format, and made online available via the supplemental information.

The noise investigations utilized two separate methods to dissect the Hall signal of the nano–tetrapods during an external field sweep. Firstly, a signal–analyzer examines the signal's PSD $S_V(f)$ in the frequency domain. This examination discloses a $1/f^2$ correlation of the PSD only when measuring the stray–field during an external magnetic field sweep inside a noise–prone region. This correlation is noteably invariant to changes in the temperature or sweeprate. A novel data acquisition technique (SR830DAQ) further scrutinizes this fluctuating nature of the nano–tetrapods' magnetic response. This SR830DAQ technique corroborates preceding findings on $1/f^2$ correlations. Additionally, sensitive fluctuations, although not detectable in a pre–amplified signal with the signal–analyzer, measured by the SR830DAQ technique, reveal a $S_V \sim I^2$ current dependence of the PSD and confirming expectations. Closer inspection of the persisting time–signal of interrupted field sweeps inside the hysteresis loop's noise–prone regions exposes metastable magnetization states with spontaneous switching processes. The author deduces that this spontaneous switching originates from thermal activation processes.

Key Points

- Best practices were pursued by fundamentally documenting and publishing both, data workflow and analysis methods. A self–written Python data analysis framework (`ana`) was created for a transparent evaluation and Continuous Analysis methods were employed to automate time–invasive tasks.
- The magnetic fingerprint of FEBID deposited three–dimensional nano–tetrapods revealed fluctuations with a characteristic $1/f^2$ behavior. These characteristics were further investigated by means of data acquisition methods. The time–signal at magnetization states with assumed complex, vortex–like magnetization discloses metastable states with spontaneous switching processes. These switching processes are concluded to arise from thermally activation.

BIBLIOGRAPHY

[1] James Clerk Maxwell. Address to the Mathematical and Physical Sections of the British Association. In *Five of Maxwell's Papers by James Clerk Maxwell*. Project Gutenberg, 1870. URL https://www.gutenberg.org/ebooks/4908.

[2] Richard P. Feynman. There's plenty of room at the bottom. *California Institute of Technology, Engineering and Science magazine*, 1960.

[3] Gordon E. Moore. Progress in digital integrated electronics [Technical literaiture, Copyright 1975 IEEE. Reprinted, with permission. Technical Digest. International Electron Devices Meeting, IEEE, 1975, pp. 11-13.]. *IEEE Solid-State Circuits Society Newsletter*, 11(3):36–37, 2006. ISSN 1098-4232. doi:10.1109/N-SSC.2006.4804410.

[4] R. Stanley Williams. What's Next? [The end of Moore's law]. *Computing in Science & Engineering*, 19(2):7–13, 2017. ISSN 1521-9615. doi:10.1109/MCSE.2017.31.

[5] Charles E. Leiserson, Neil C. Thompson, Joel S. Emer, Bradley C. Kuszmaul, Butler W. Lampson, Daniel Sanchez, and Tao B. Schardl. There's plenty of room at the Top: What will drive computer performance after Moore's law? *Science*, 368(6495), 2020. ISSN 1095-9203. doi:10.1126/science.aam9744.

[6] D. N. Basov, R. D. Averitt, and D. Hsieh. Towards properties on demand in quantum materials. *Nature materials*, 16(11):1077–1088, 2017. ISSN 1476-1122. doi:10.1038/nmat5017.

[7] Yoshinori Tokura, Masashi Kawasaki, and Naoto Nagaosa. Emergent functions of quantum materials. *Nature Physics*, 13(11):1056–1068, 2017. ISSN 1745-2473. doi:10.1038/nphys4274.

[8] Yongbing Xu, David D. Awschalom, and Junsaku Nitta. *Handbook of Spintronics*. Springer Netherlands, Dordrecht, 2016. ISBN 978-94-007-6891-8. doi:10.1007/978-94-007-6892-5.

[9] Katsuaki Sato, Eiji Saitoh, Arthur Willoughby, Peter Capper, and Safa Kasap. *Spintronics for Next Generation Innovative Devices*. Wiley, Chicester, 1st ed. edition, 2015. ISBN 9781118751916.

[10] Fabio Pulizzi. Spintronics. *Nature materials*, 11(5):367, 2012. ISSN 1476-1122. doi:10.1038/nmat3327.

[11] Claude Chappert, Albert Fert, and Frédéric Nguyen van Dau. The emergence of spin electronics in data storage. *Nature materials*, 6(11):813–823, 2007. ISSN 1476-1122. doi:10.1038/nmat2024.

[12] Guilherme Migliato Marega, Yanfei Zhao, Ahmet Avsar, Zhenyu Wang, Mukesh Tripathi, Aleksandra Radenovic, and Andras Kis. Logic-in-memory based on an atomically thin semiconductor. *Nature*, 587(7832):72–77, 2020. ISSN 1476-4687. doi:10.1038/s41586-020-2861-0.

[13] Abu Sebastian, Manuel Le Gallo, Riduan Khaddam-Aljameh, and Evangelos Eleftheriou. Memory devices and applications for in-memory computing. *Nature nanotechnology*, 15(7):529–544, 2020. doi:10.1038/s41565-020-0655-z.

[14] Manuel Le Gallo, Abu Sebastian, Roland Mathis, Matteo Manica, Heiner Giefers, Tomas Tuma, Costas Bekas, Alessandro Curioni, and Evangelos Eleftheriou. Mixed-precision in-memory computing. *Nature Electronics*, 1(4):246–253, 2018. ISSN 2520-1131. doi:10.1038/s41928-018-0054-8.

[15] Youhui Zhang, Peng Qu, Yu Ji, Weihao Zhang, Guangrong Gao, Guanrui Wang, Sen Song, Guoqi Li, Wenguang Chen, Weimin Zheng, Feng Chen, Jing Pei, Rong Zhao, Mingguo Zhao, and Luping Shi. A system hierarchy for brain-inspired computing. *Nature*, 586(7829):378–384, 2020. ISSN 1476-4687. doi:10.1038/s41586-020-2782-y.

[16] Kaushik Roy, Akhilesh Jaiswal, and Priyadarshini Panda. Towards spike-based machine intelligence with neuromorphic computing. *Nature*, 575(7784): 607–617, 2019. ISSN 1476-4687. doi:10.1038/s41586-019-1677-2.

[17] Udo Flohr. Sensorik: Laserscanner jetzt auch in klein. *Technology Review*, 12/2020:8, 2020-04-11. URL https://www.heise.de/select/tr/2020/12/2028014285986777434.

[18] R. Skomski. Nanomagnetics. *Journal of Physics: Condensed Matter*, 15(20): R841–R896, 2003. ISSN 0953-8984. doi:10.1088/0953-8984/15/20/202.

[19] Chunsen Liu, Huawei Chen, Shuiyuan Wang, Qi Liu, Yu-Gang Jiang, David Wei Zhang, Ming Liu, and Peng Zhou. Two-dimensional materials for next-generation computing technologies. *Nature nanotechnology*, 15(7): 545–557, 2020. doi:10.1038/s41565-020-0724-3.

[20] Amalio Fernández-Pacheco, Robert Streubel, Olivier Fruchart, Riccardo Hertel, Peter Fischer, and Russell P. Cowburn. Three-dimensional nanomagnetism. *Nature communications*, 8:15756, 2017. doi:10.1038/ncomms15756.

[21] Anjan Barman, Sucheta Mondal, Sourav Sahoo, and Anulekha. Magnetization Dynamics of Nanoscale Magnetic Materials: A Perspective. URL `https://arxiv.org/pdf/2008.05819`.

[22] Yoshinori Tokura and Naoya Kanazawa. Magnetic Skyrmion Materials. *Chemical reviews*, 2020. doi:10.1021/acs.chemrev.0c00297.

[23] Hans-Benjamin Braun, Remo Hügli, and Laura. Heyderman. Nordpole ohne Südpole. *Spektrum der Wissenschaft*, 2/2014, 2014. URL `https://www.spektrum.de/artikel/1216441`.

[24] Albert Fert, Vincent Cros, and João Sampaio. Skyrmions on the track. *Nature nanotechnology*, 8(3):152–156, 2013. doi:10.1038/nnano.2013.29.

[25] L. D. C. Jaubert and P. C. W. Holdsworth. Magnetic monopole dynamics in spin ice. *Journal of physics. Condensed matter : an Institute of Physics journal*, 23(16):164222, 2011. doi:10.1088/0953-8984/23/16/164222. URL `https://iopscience.iop.org/article/10.1088/0953-8984/23/16/164222`.

[26] Johannes Bausch, Toby S. Cubitt, and James D. Watson. Uncomputability of phase diagrams. *Nature Communications*, 12(1):452, 2021. ISSN 2041-1723. doi:10.1038/s41467-020-20504-6. URL `https://www.nature.com/articles/s41467-020-20504-6`.

[27] Devon M. Simmonds. The Programming Paradigm Evolution. *Computer*, 45 (6):93–95, 2012. ISSN 0018-9162. doi:10.1109/MC.2012.219.

[28] T.-Y.B. Yang, G. Furnish, and P. F. Dubois. Steering object-oriented scientific computations. In Raimund K. Ege, Madhu Singh, and Bertrand Meyer, editors, *Technology of object-oriented languages and systems*, pages 112–119, Los Alamitos, Calif. u.a, 1997. IEEE Computer Society. ISBN 0-8186-8383-X. doi:10.1109/TOOLS.1997.654712.

[29] Masaharu Goto, Andrei Gheata, Mihaela Gheata, Olivier Couet, Ilka Antcheva, Bertrand Bellenot, Valeriy Onoutchin, Gerri Ganis, Maarten Ballintijn, Valerie Fine, Victor Perevoztchikov, Nenad Buncic, Suzanne Panacek, Axel Naumann, Anna Kreshuk, Richard Maunder, Timur Pocheptsov, Sergei Linev, Stefan Roiser, Lorenzo Moneta, and Wim Lavrijsen. ROOT User‘s Guide, 2018. URL https://root.cern/root/htmldoc/guides/users-guide/ROOTUsersGuide.html.

[30] UC Berkeley. What is Data Science?, 23.07.2020. URL https://datascience.berkeley.edu/about/what-is-data-science/.

[31] Changing the culture of data science, 24.07.2020. URL https://www.turing.ac.uk/research/impact-stories/changing-culture-data-science.

[32] Nature Research group. Harnessing the power of computational science: Collection, 2020. URL https://www.nature.com/collections/adcghheeaj.

[33] Elizabeth Hawkins. A dedicated home for computational science : Of Schemes and Memes Blog, 2020. URL http://blogs.nature.com/ofschemesandmemes/2020/04/27/a-dedicated-home-for-computational-science.

[34] Nature Research group. Scientific Data. URL https://www.nature.com/sdata/.

[35] Center for Open Science. Homepage, 14.11.2020. URL https://www.cos.io/.

[36] Xiaoli Chen, Sünje Dallmeier-Tiessen, Robin Dasler, Sebastian Feger, Pamfilos Fokianos, Jose Benito Gonzalez, Harri Hirvonsalo, Dinos Kousidis, Artemis Lavasa, Salvatore Mele, Diego Rodriguez Rodriguez, Tibor Šimko, Tim Smith, Ana Trisovic, Anna Trzcinska, Ioannis Tsanaktsidis, Markus Zimmermann, Kyle Cranmer, Lukas Heinrich, Gordon Watts, Michael Hildreth, Lara Lloret Iglesias, Kati Lassila-Perini, and Sebastian Neubert. Open is not enough. *Nature Physics*, 15(2):113–119, 2019. ISSN 1745-2473. doi:10.1038/s41567-018-0342-2.

[37] Boris Pritychenko. The value of archived data. *Nature Reviews Physics*, 2(5): 224–225, 2020. doi:10.1038/s42254-020-0173-9.

[38] Julia S. Stewart Lowndes, Benjamin D. Best, Courtney Scarborough, Jamie C. Afflerbach, Melanie R. Frazier, Casey C. O'Hara, Ning Jiang, and Benjamin S. Halpern. Our path to better science in less time using open data science tools. *Nature ecology & evolution*, 1(6):160, 2017. doi:10.1038/s41559-017-0160.

[39] Victoria Stodden. The data science life cycle. *Communications of the ACM*, 63(7):58–66, 2020. ISSN 0001-0782. doi:10.1145/3360646.

[40] David Bailey, Jonathan Borwein, Randall LeVeque, Bill Rider, William Stein, and Victoria Stodden. ICERM - Reproducibility in Computational and Experimental Mathematics, 2013. URL `https://icerm.brown.edu/topical_workshops/tw12-5-rcem/`.

[41] Alyssa Goodman, Alberto Pepe, Alexander W. Blocker, Christine L. Borgman, Kyle Cranmer, Merce Crosas, Rosanne Di Stefano, Yolanda Gil, Paul Groth, Margaret Hedstrom, David W. Hogg, Vinay Kashyap, Ashish Mahabal, Aneta Siemiginowska, and Aleksandra Slavkovic. Ten simple rules for the care and feeding of scientific data. *PLoS computational biology*, 10(4):e1003542, 2014. doi:10.1371/journal.pcbi.1003542.

[42] Victoria Stodden and Sheila Miguez. Best Practices for Computational Science: Software Infrastructure and Environments for Reproducible and Extensible Research. *Journal of Open Research Software*, 2(1), 2014. ISSN 2049-9647. doi:10.5334/jors.ay.

[43] Greg Wilson, Jennifer Bryan, Karen Cranston, Justin Kitzes, Lex Nederbragt, and Tracy K. Teal. Good Enough Practices in Scientific Computing. *PLoS computational biology*, 13(6):e1005510, 2016 // 2017. doi:10.1371/journal.pcbi.1005510. URL `https://arxiv.org/pdf/1609.00037`.

[44] Nick Barnes, David Jones, Peter Norvig, Cameron Neylon, Rufus Pollock, Joseph Jackson, Victoria Stodden, and Peter Suber. Science Code Manifesto, 24.07.2020. URL `http://sciencecodemanifesto.org/`.

[45] Mark D. Wilkinson, Michel Dumontier, I. Jsbrand Jan Aalbersberg, Gabrielle Appleton, Myles Axton, Arie Baak, Niklas Blomberg, Jan-Willem Boiten, Luiz Bonino da Silva Santos, Philip E. Bourne, Jildau Bouwman, Anthony J. Brookes, Tim Clark, Mercè Crosas, Ingrid Dillo, Olivier Dumon, Scott Edmunds, Chris T. Evelo, Richard Finkers, Alejandra Gonzalez-Beltran, Alasdair J. G. Gray, Paul Groth, Carole Goble, Jeffrey S. Grethe, Jaap Heringa, Peter A. C. 't Hoen, Rob Hooft, Tobias Kuhn, Ruben Kok, Joost Kok, Scott J. Lusher, Maryann E. Martone, Albert Mons, Abel L. Packer, Bengt Persson, Philippe Rocca-Serra, Marco Roos, Rene van Schaik, Susanna-Assunta Sansone, Erik Schultes, Thierry Sengstag, Ted Slater, George Strawn, Morris A. Swertz, Mark Thompson, Johan van der Lei, Erik van Mulligen, Jan Velterop, Andra Waagmeester, Peter Wittenburg, Katherine Wolstencroft, Jun Zhao, and Barend Mons. The FAIR Guiding Principles for sci-

entific data management and stewardship. *Scientific data*, 3:160018, 2016. doi:10.1038/sdata.2016.18.

[46] GO FAIR. FAIR Principles - GO FAIR, 18.08.2020. URL https://www.go-fair.org/fair-principles/.

[47] Research Data Alliance FAIR Data Maturity Model Working Group. FAIR Data Maturity Model: specification and guidelines.

[48] European Commission. European Open Science Cloud (EOSC) Declaration, 2017. URL https://ec.europa.eu/research/openscience/index.cfm?pg=open-science-cloud.

[49] European Commission. *Turning FAIR data into reality: Final report and action plan from the European Commission expert group on FAIR data*. Publications Office of the European Union, Luxembourg, 2018. ISBN 978-92-79-96546-3. doi:10.2777/1524.

[50] European Commission. Open Science, 2019. URL https://ec.europa.eu/research/openscience/index.cfm.

[51] Jeffrey M. Perkel. Make code accessible with these cloud services. *Nature*, 575(7781):247–248, 2019. ISSN 1476-4687. doi:10.1038/d41586-019-03366-x. URL https://www.nature.com/articles/d41586-019-03366-x.

[52] Monya Baker. Why scientists must share their research code. *Nature*, 2016. ISSN 1476-4687. doi:10.1038/nature.2016.20504.

[53] A. Morin, J. Urban, P. D. Adams, I. Foster, A. Sali, D. Baker, and P. Sliz. Research priorities. Shining light into black boxes. *Science*, 336(6078):159–160, 2012. ISSN 1095-9203. doi:10.1126/science.1218263. URL https://science.sciencemag.org/content/336/6078/159.

[54] Nick Barnes. Publish your computer code: it is good enough. *Nature*, 467 (7317):753, 2010. ISSN 1476-4687. doi:10.1038/467753a.

[55] Deutsche Forschungsgemeinschaft. Guidelines for Safeguarding Good Research Practice. Code of Conduct. 2019. doi:10.5281/ZENODO.3923602.

[56] National Science Foundation. Social, Behavioral, and Economic Sciences Perspectives on Robust and Reliable Science: Report of the Subcommittee on Replicability in Science, 2015. URL https://www.nsf.gov/sbe/AC_Materials/SBE_Robust_and_Reliable_Research_Report.pdf.

[57] National Science Foundation. Grant General Conditions (GC-1), 2020. URL https://www.nsf.gov/awards/managing/general_conditions.jsp.

[58] IEEE Recommended Practice for Software Requirements Specifications, 1998-10-20. URL https://doi.org/10.1109/IEEESTD.1998.88286.

[59] National Science Foundation. Dissemination and Sharing of Research Results, 28.10.2020. URL https://www.nsf.gov/bfa/dias/policy/dmp.jsp.

[60] Victoria Stodden. The Legal Framework for Reproducible Scientific Research: Licensing and Copyright. *Computing in Science & Engineering*, 11(1):35–40, 2009. ISSN 1521-9615. doi:10.1109/MCSE.2009.19.

[61] Open Source Initiative. Open Source Licenses by Category, 21.07.2020. URL https://opensource.org/licenses/category.

[62] Choose a License. Appendix: Choose a License, 23.7.2020. URL https://choosealicense.com/appendix/.

[63] Choose a License. Licenses, 23.7.2020. URL https://choosealicense.com/licenses/.

[64] Jeffrey M. Perkel. Challenge to scientists: does your ten-year-old code still run? *Nature*, 584(7822):656–658, 2020. ISSN 1476-4687. doi:10.1038/d41586-020-02462-7. URL https://www.nature.com/articles/d41586-020-02462-7.

[65] Jeffrey M. Perkel. A toolkit for data transparency takes shape. *Nature*, 560 (7719):513–515, 2018. ISSN 1476-4687. doi:10.1038/d41586-018-05990-5.

[66] Amanda Casari, Katie McLaughlin, Milo Z. Trujillo, Jean-Gabriel Young, James P. Bagrow, and Laurent Hébert-Dufresne. Open source ecosystems need equitable credit across contributions. *Nature Computational Science*, 1 (1):2, 2021. doi:10.1038/s43588-020-00011-w.

[67] Alex O. Holcombe. Contributorship, Not Authorship: Use CRediT to Indicate Who Did What. *Publications*, 7(3):48, 2019. doi:10.3390/publications7030048. URL https://www.mdpi.com/2304-6775/7/3/48.

[68] Open Source Initiative. The Open Source Definition, 2007-03-22. URL https://opensource.org/osd.

[69] Thomas Pasquier, Matthew K. Lau, Ana Trisovic, Emery R. Boose, Ben Couturier, Mercè Crosas, Aaron M. Ellison, Valerie Gibson, Chris R. Jones, and

Margo Seltzer. If these data could talk. *Scientific data*, 4:170114, 2017. doi:10.1038/sdata.2017.114.

[70] Kristina Maria Hettne, Peter Verhaar, Erik Schultes, and Laurents Sesink. From FAIR Leading Practices to FAIR Implementation and Back: An Inclusive Approach to FAIR at Leiden University Libraries. *Data Science Journal*, 19, 2020. doi:10.5334/dsj-2020-040.

[71] Harvard University. Data Lifecycle, 07.11.2020. URL `https://researchdatamanagement.harvard.edu/data-lifecycle`.

[72] University of Reading. Open Research Handbook: Reproducibility, 24.07.2020. URL `https://libguides.reading.ac.uk/open-research/reproducibility`.

[73] Charles Q. Choi. Migrating big astronomy data to the cloud. *Nature*, 584 (7819):159–160, 2020. ISSN 1476-4687. doi:10.1038/d41586-020-02284-7. URL `https://www.nature.com/articles/d41586-020-02284-7`.

[74] Padraig Gleeson, Andrew P. Davison, R. Angus Silver, and Giorgio A. Ascoli. A Commitment to Open Source in Neuroscience. *Neuron*, 96(5):964–965, 2017. doi:10.1016/j.neuron.2017.10.013.

[75] Eilif Muller, James A. Bednar, Markus Diesmann, Marc-Oliver Gewaltig, Michael Hines, and Andrew P. Davison. Python in neuroscience. *Frontiers in neuroinformatics*, 9:11, 2015. ISSN 1662-5196. doi:10.3389/fninf.2015.00011.

[76] Mark Hahnel, Jon Treadway, Briony Fane, Robert Kiley, Dale Peters, and Grace Baynes. The State of Open Data Report 2017: Digital Science Report.

[77] Christophe Bahim, Carlos Casorrán-Amilburu, Makx Dekkers, Edit Herczog, Nicolas Loozen, Konstantinos Repanas, Keith Russell, and Shelley Stall. The FAIR Data Maturity Model: An Approach to Harmonise FAIR Assessments. *Data Science Journal*, 19, 2020. doi:10.5334/dsj-2020-041.

[78] Massimiliano Assante, Leonardo Candela, Donatella Castelli, and Alice Tani. Are Scientific Data Repositories Coping with Research Data Publishing? *Data Science Journal*, 15, 2016. doi:10.5334/dsj-2016-006.

[79] Salvador García, Sergio Ramírez-Gallego, Julián Luengo, José Manuel Benítez, and Francisco Herrera. Big data preprocessing: methods and prospects. *Big Data Analytics*, 1(1), 2016. doi:10.1186/s41044-016-0014-0.

[80] Balazs Aczel, Barnabas Szaszi, Alexandra Sarafoglou, Zoltan Kekecs, Šimon

Kucharský, Daniel Benjamin, Christopher D. Chambers, Agneta Fisher, Andrew Gelman, Morton A. Gernsbacher, John P. Ioannidis, Eric Johnson, Kai Jonas, Stavroula Kousta, Scott O. Lilienfeld, D. Stephen Lindsay, Candice C. Morey, Marcus Munafò, Benjamin R. Newell, Harold Pashler, David R. Shanks, Daniel J. Simons, Jelte M. Wicherts, Dolores Albarracin, Nicole D. Anderson, John Antonakis, Hal R. Arkes, Mitja D. Back, George C. Banks, Christopher Beevers, Andrew A. Bennett, Wiebke Bleidorn, Ty W. Boyer, Cristina Cacciari, Alice S. Carter, Joseph Cesario, Charles Clifton, Ronán M. Conroy, Mike Cortese, Fiammetta Cosci, Nelson Cowan, Jarret Crawford, Eveline A. Crone, John Curtin, Randall Engle, Simon Farrell, Pasco Fearon, Mark Fichman, Willem Frankenhuis, Alexandra M. Freund, M. Gareth Gaskell, Roger Giner-Sorolla, Don P. Green, Robert L. Greene, Lisa L. Harlow, Fernando Hoces de La Guardia, Derek Isaacowitz, Janet Kolodner, Debra Lieberman, Gordon D. Logan, Wendy B. Mendes, Lea Moersdorf, Brendan Nyhan, Jeffrey Pollack, Christopher Sullivan, Simine Vazire, and Eric-Jan Wagenmakers. A consensus-based transparency checklist. *Nature Human Behaviour*, 4 (1):4–6, 2020. ISSN 2397-3374. doi:10.1038/s41562-019-0772-6.

[81] Brooks Hanson, Andrew Sugden, and Bruce Alberts. Making data maximally available. *Science*, 331(6018):649, 2011. ISSN 1095-9203. doi:10.1126/science.1203354. URL https://science.sciencemag.org/content/331/6018/649.

[82] Dealing with data. Challenges and opportunities. Introduction. *Science*, 331 (6018):692–693, 2011. ISSN 1095-9203. doi:10.1126/science.331.6018.692.

[83] Patrick Vandewalle. Code Sharing Is Associated with Research Impact in Image Processing. *Computing in Science & Engineering*, 14(4):42–47, 2012. ISSN 1521-9615. doi:10.1109/MCSE.2012.63.

[84] Jane S. Greaves, Anita M. S. Richards, William Bains, Paul B. Rimmer, Hideo Sagawa, David L. Clements, Sara Seager, Janusz J. Petkowski, Clara Sousa-Silva, Sukrit Ranjan, Emily Drabek-Maunder, Helen J. Fraser, Annabel Cartwright, Ingo Mueller-Wodarg, Zhuchang Zhan, Per Friberg, Iain Coulson, E'lisa Lee, and Jim Hoge. Phosphine gas in the cloud decks of Venus. *Nature Astronomy*, 2020. doi:10.1038/s41550-020-1174-4.

[85] I. A. G. Snellen, L. Guzman-Ramirez, M. R. Hogerheijde, A. P. S. Hygate, and F. F. S. van der Tak. Re-analysis of the 267-GHz ALMA observations of Venus: No statistically significant detection of phosphine. URL https://arxiv.org/pdf/2010.09761v1.

[86] Alexandra Witze. Life on Venus claim faces strongest challenge yet. *Nature*, 590(7844):19–20, 2021. ISSN 1476-4687. doi:10.1038/d41586-021-00249-y.

[87] Library Carpentry. Top 10 FAIR Data & Software Things, 28.06.2020. URL https://librarycarpentry.org/Top-10-FAIR/.

[88] terms4fairskills.github.io. A terminology for FAIR Stewardship Skills Workshop, 20.09.2020. URL https://terms4fairskills.github.io/.

[89] Kathleen Gregory. A dataset describing data discovery and reuse practices in research. *Scientific data*, 7(1):232, 2020. doi:10.1038/s41597-020-0569-5.

[90] Elsevier. Open data report, 22.07.2020. URL https://www.elsevier.com/open-science/research-data/open-data-report.

[91] Richard C. Jennings. Data selection and responsible conduct: was Millikan a fraud? *Science and Engineering Ethics*, 10(4):639–653, 2004. ISSN 1471-5546. doi:10.1007/s11948-004-0044-2.

[92] GAIA-X. A Federated Data Infrastructure for Europe, 28.01.2021. URL https://www.data-infrastructure.eu.

[93] European Commission. Commission proposes measures to boost data sharing, 2020-11-25. URL https://ec.europa.eu/commission/presscorner/detail/en/ip_20_2102.

[94] Edison Project. EDISON Data Science Framework (EDSF), 2017. URL https://edison-project.eu/edison/edison-data-science-framework-edsf/.

[95] European Commission. Facts and Figures for open research data, 2019. URL https://ec.europa.eu/info/research-and-innovation/strategy/goals-research-and-innovation-policy/open-science/open-science-monitor/facts-and-figures-open-research-data_en.

[96] Datenethikkommission. Gutachten der Datenethikkommission, 18.10.2020. URL https://datenethikkommission.de/gutachten/.

[97] Stefan Krempl. Datenstrategie: Bundesregierung will Potenzial von Daten erschließen. *heise Online*, 2021-01-27. URL https://www.heise.de/news/Datenstrategie-Bundesregierung-will-Potenzial-von-Daten-erschliessen-5038078.html.

[98] Jess Whittlestone, Rune Nyrup, Anna Alexandrova, Kanta Dihal, and Stephen Cave. *Ethical and societal implications of algorithms, data, and artificial intelligence: a roadmap for research*. 2019. URL https://www.researchgate.net/publication/337565648_Ethical_

and_societal_implications_of_algorithms_data_and_artificial_intelligence_a_roadmap_for_research.

[99] Markus Feilner. Jetzt oder nie! Digitale Selbstbestimmung als Schlüssel zur Freiheit. *iX*, 12/2020:42, 2020-11-18. URL https://www.heise.de/select/ix/2020/12/2031222112643317938.

[100] Nature Research group. Challenges in irreproducible research: Collection, 2018. URL https://www.nature.com/collections/prbfkwmwvz.

[101] Must try harder. *Nature*, 483(7391):509, 2012. ISSN 1476-4687. doi:10.1038/483509a.

[102] Marcia McNutt. Reproducibility. *Science*, 343(6168):229, 2014. ISSN 1095-9203. doi:10.1126/science.1250475.

[103] Stuart Buck. Solving reproducibility. *Science*, 348(6242):1403, 2015. ISSN 1095-9203. doi:10.1126/science.aac8041.

[104] Monya Baker. 1,500 scientists lift the lid on reproducibility. *Nature*, 533(7604): 452–454, 2016. ISSN 1476-4687. doi:10.1038/533452a.

[105] Jon Brock. "It's not a replication crisis. It's an innovation opportunity", 28 October 2019. URL https://www.natureindex.com/news-blog/not-a-replication-crisis-innovation-opportunity.

[106] Ulrich Dirnagl. Metaforschung: Kulturwandel in der Biomedizin. *Spektrum der Wissenschaft*, 10:38–43, 2020. URL https://www.spektrum.de/artikel/1760432.

[107] C. Glenn Begley and Lee M. Ellis. Drug development: Raise standards for preclinical cancer research. *Nature*, 483(7391):531–533, 2012. ISSN 1476-4687. doi:10.1038/483531a.

[108] Gemma Conroy. The 7 deadly sins of research, 2019. URL https://www.natureindex.com/news-blog/the-seven-deadly-sins-of-research.

[109] Jeffrey T. Leek and Leah R. Jager. Is Most Published Research Really False? *Annual Review of Statistics and Its Application*, 4(1):109–122, 2017. ISSN 2326-8298. doi:10.1146/annurev-statistics-060116-054104.

[110] Jeff Leek, Blakeley B. McShane, Andrew Gelman, David Colquhoun, Michèle B. Nuijten, and Steven N. Goodman. Five ways to fix statistics.

Nature, 551(7682):557–559, 2017. ISSN 1476-4687. doi:10.1038/d41586-017-07522-z.

[111] Michèle B. Nuijten, Chris H. J. Hartgerink, Marcel A. L. M. van Assen, Sacha Epskamp, and Jelte M. Wicherts. The prevalence of statistical reporting errors in psychology (1985-2013). *Behavior Research Methods*, 48(4):1205–1226, 2016. ISSN 1554-3528. doi:10.3758/s13428-015-0664-2.

[112] Jeffrey T. Leek and Roger D. Peng. Statistics: P values are just the tip of the iceberg. *Nature*, 520(7549):612, 2015. ISSN 1476-4687. doi:10.1038/520612a.

[113] Regina Nuzzo. Scientific method: statistical errors. *Nature*, 506(7487):150–152, 2014. ISSN 1476-4687. doi:10.1038/506150a.

[114] Zeeya Merali. Computational science: ...Error. *Nature*, 467(7317):775–777, 2010. ISSN 1476-4687. doi:10.1038/467775a.

[115] Konrad Hinsen. [Rp] Stokes drag on conglomerates of spheres. *ReScience C*, 6(1), 2020. doi:10.5281/ZENODO.3889694. URL `https://zenodo.org/record/3889694`.

[116] Konrad Hinsen. [Rp] Structural flexibility in proteins - impact of the crystal environment. *ReScience C*, 6(1), 2020. doi:10.5281/ZENODO.3886447. URL `https://zenodo.org/record/3886447`.

[117] Jeffrey M. Perkel. Workflow systems turn raw data into scientific knowledge. *Nature*, 573(7772):149–150, 2019. ISSN 1476-4687. doi:10.1038/d41586-019-02619-z.

[118] Emil Bjornson. Reproducible Research: Best Practices and Potential Misuse. *IEEE Signal Processing Magazine*, 36(3):106–123, 2019. ISSN 1053-5888. doi:10.1109/MSP.2019.2898421.

[119] Geir Kjetil Sandve, Anton Nekrutenko, James Taylor, and Eivind Hovig. Ten simple rules for reproducible computational research. *PLoS computational biology*, 9(10):e1003285, 2013. doi:10.1371/journal.pcbi.1003285.

[120] William Stafford Noble. A quick guide to organizing computational biology projects. *PLoS computational biology*, 5(7):e1000424, 2009. doi:10.1371/journal.pcbi.1000424.

[121] Victoria Stodden. Beyond Open Data: A Model for Linking Digital Artifacts to Enable Reproducibility of Scientific Claims. In Ivo Jimenez, Carlos Maltzahn, and Jay Lofstead, editors, *Proceedings of the 3rd International*

Workshop on Practical Reproducible Evaluation of Computer Systems, pages 9–14, [S.l.], 2020. Association for Computing Machinery. ISBN 9781450379779. doi:10.1145/3391800.3398172.

[122] Edward Raff. Quantifying Independently Reproducible Machine Learning. *The Gradient*, 2020. URL `https://thegradient.pub/independently-reproducible-machine-learning/`.

[123] Helen Shen. Interactive notebooks: Sharing the code. *Nature*, 515(7525): 151–152, 2014. ISSN 1476-4687. doi:10.1038/515151a.

[124] Lorena A. Barba. Reproducibility PI Manifesto, 2012. URL `https://figshare.com/articles/reproducibility_pi_manifesto/104539`.

[125] Victoria Stodden. Reproducible Research: Tools and Strategies for Scientific Computing. *Computing in Science & Engineering*, 14(4):11–12, 2012. ISSN 1521-9615. doi:10.1109/MCSE.2012.82.

[126] David L. Donoho, Arian Maleki, Inam Ur Rahman, Morteza Shahram, and Victoria Stodden. Reproducible Research in Computational Harmonic Analysis. *Computing in Science & Engineering*, 11(1):8–18, 2009. ISSN 1521-9615. doi:10.1109/MCSE.2009.15.

[127] G. Wilson. Software Carpentry: Getting Scientists to Write Better Code by Making Them More Productive. *Computing in Science & Engineering*, 8(6): 66–69, 2006. ISSN 1521-9615. doi:10.1109/MCSE.2006.122.

[128] M. Schwab, N. Karrenbach, and J. Claerbout. Making scientific computations reproducible. *Computing in Science & Engineering*, 2(6):61–67, 2000. ISSN 1521-9615. doi:10.1109/5992.881708.

[129] Denis Stalz-John Marcus Hanhart. Viele Vorzüge: Python und maschinelles Lernen. *iX*, 1/2021:46–49, 2020-12-16. URL `https://www.heise.de/select/ix/2021/1/2026911212863502355`.

[130] Fernando Perez, Brian E. Granger, and John D. Hunter. Python: An Ecosystem for Scientific Computing. *Computing in Science & Engineering*, 13(2): 13–21, 2011. ISSN 1521-9615. doi:10.1109/MCSE.2010.119.

[131] Wes McKinney. *Python for data analysis: Data wrangling with Pandas, NumPy, and IPython*. O'Reilly Media, Sebastopol, CA, second edition edition, 2017. ISBN 9781491957660. URL `https://wesmckinney.com/pages/book.html`.

[132] Jacob T. Vanderplas. *Python data science handbook: Essential tools for working with data*. O'Reilly Media Inc, Sebastopol, CA, first edition edition, 2016. ISBN 9781491912140. URL https://jakevdp.github.io/PythonDataScienceHandbook/.

[133] Stefan Behnel, Robert Bradshaw, Craig Citro, Lisandro Dalcin, Dag Sverre Seljebotn, and Kurt Smith. Cython: The Best of Both Worlds. *Computing in Science & Engineering*, 13(2):31–39, 2011. ISSN 1521-9615. doi:10.1109/MCSE.2010.118.

[134] Wes McKinney. Data Structures for Statistical Computing in Python. In *Proceedings of the 9th Python in Science Conference*. SciPy, 2010. doi:10.25080/Majora-92bf1922-00a.

[135] Nico Kreiling. Stein auf Stein: Ein Streifzug durch die PyData-IT-Landschaft. *iX special*, 15:34–39, 2020. URL https://www.heise.de/select/ix-special/archiv/2020/15/seite-34.

[136] Charles R. Harris, K. Jarrod Millman, Stéfan J. van der Walt, Ralf Gommers, Pauli Virtanen, David Cournapeau, Eric Wieser, Julian Taylor, Sebastian Berg, Nathaniel J. Smith, Robert Kern, Matti Picus, Stephan Hoyer, Marten H. van Kerkwijk, Matthew Brett, Allan Haldane, Jaime Fernández Del Río, Mark Wiebe, Pearu Peterson, Pierre Gérard-Marchant, Kevin Sheppard, Tyler Reddy, Warren Weckesser, Hameer Abbasi, Christoph Gohlke, and Travis E. Oliphant. Array programming with NumPy. *Nature*, 585(7825):357–362, 2020. ISSN 1476-4687. doi:10.1038/s41586-020-2649-2.

[137] Pauli Virtanen, Ralf Gommers, Travis E. Oliphant, Matt Haberland, Tyler Reddy, David Cournapeau, Evgeni Burovski, Pearu Peterson, Warren Weckesser, Jonathan Bright, Stéfan J. van der Walt, Matthew Brett, Joshua Wilson, K. Jarrod Millman, Nikolay Mayorov, Andrew R. J. Nelson, Eric Jones, Robert Kern, Eric Larson, C. J. Carey, İlhan Polat, Yu Feng, Eric W. Moore, Jake VanderPlas, Denis Laxalde, Josef Perktold, Robert Cimrman, Ian Henriksen, E. A. Quintero, Charles R. Harris, Anne M. Archibald, Antônio H. Ribeiro, Fabian Pedregosa, and Paul van Mulbregt. SciPy 1.0: fundamental algorithms for scientific computing in Python. *Nature Methods*, 17 (3):261–272, 2020. ISSN 1548-7105. doi:10.1038/s41592-019-0686-2. URL https://www.nature.com/articles/s41592-019-0686-2.

[138] Stéfan van der Walt, S. Chris Colbert, and Gaël Varoquaux. The NumPy Array: A Structure for Efficient Numerical Computation. *Computing in Science & Engineering*, 13(2):22–30, 2011. ISSN 1521-9615. doi:10.1109/MCSE.2011.37.

[139] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, Olivier Grisel, Mathieu Blondel, Peter Prettenhofer, Ron Weiss, Vincent Dubourg, Jake VanderPlas, Alexandre Passos, David Cournapeau, Matthieu Brucher, Matthieu Perrot, and Édouard Duchesnay. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research*, 12(85):2825–2830, 2011. ISSN 1533-7928.

[140] Fernando Perez and Brian E. Granger. IPython: A System for Interactive Scientific Computing. *Computing in Science & Engineering*, 9(3):21–29, 2007. ISSN 1521-9615. doi:10.1109/MCSE.2007.53.

[141] John D. Hunter. Matplotlib: A 2D Graphics Environment. *Computing in Science & Engineering*, 9(3):90–95, 2007. ISSN 1521-9615. doi:10.1109/MCSE.2007.55.

[142] Georg Brandl. Sphinx: Python Documentation Generator, 2007. URL `http://sphinx-doc.org/`.

[143] GitLab. The first single application for the entire DevOps lifecycle, 2020. URL `https://about.gitlab.com/`.

[144] GitHub. A development platform to host and review code, manage projects, and build software, 2020. URL `https://github.com/`.

[145] Git Team. Git Version Control System. URL `https://git-scm.com/`.

[146] Jon Loeliger. *Version control with Git: Powerful tools and techniques for collaborative software development*. Safari Tech Books Online. O'Reilly, Beijing, 1st ed. edition, 2009. ISBN 978-0-596-52012-0. URL `http://gbv.eblib.com/patron/FullRecord.aspx?p=443377`.

[147] Susan Potter. Git. In Amy Brown and Greg Wilson, editors, *The Architecture of Open Source Applications*, volume II of *The architecture of open source applications*. AOSA, Erscheinungsort nicht ermittelbar, 2012? ISBN 9781105571817. URL `https://www.aosabook.org/en/git.html`.

[148] C. Titus Brown and Rosangela Canino-Koning. Continuous Integration, 29.06.2016. URL `https://www.aosabook.org/en/integration.html`.

[149] Moritz Lenz. *Python Continuous Integration and Delivery: A Concise Guide with Examples*. Apress, Berkeley, CA, 2019. ISBN 9781484242827. doi:10.1007/978-1-4842-4281-0.

[150] Carl Boettiger. An introduction to Docker for reproducible research. *ACM*

SIGOPS Operating Systems Review, 49(1):71–79, 2015. ISSN 0163-5980. doi:10.1145/2723872.2723882.

[151] Michael Friedrich. Quo vadis, DevOps? GitLab CI und CD installieren und konfigurieren. *iX*, 2019(4):118, 2019-03-20. URL `https://ix.de/ix1904118`.

[152] Jonas Hecht. GitLab mit passender Continuous-Integration-Pipeline automatisiert aufsetzen. *heise Online*, 2018-11-26. URL `https://heise.de/-4224990`.

[153] Brett K. Beaulieu-Jones and Casey S. Greene. Reproducibility of computational workflows is automated using continuous analysis. *Nature biotechnology*, 35(4):342–346, 2017. doi:10.1038/nbt.3780.

[154] Urheberrechtsgesetz, 11.11.2020. URL `https://www.urheberrecht.org/law/normen/urhg/2016-12-20/text/`.

[155] Creative Commons. Share your work, 26.04.2019. URL `https://creativecommons.org/share-your-work/`.

[156] Choose a License. MIT License, 23.7.2020.

[157] Mihai Barbulescu, Mariana Marinescu, Viorel Marinescu, Oana Grigoriu, Giorgian Neculoiu, Virginia Sandulescu, and Ionela Halcu. GNU GPL in studying programs from the Systems Engineering field. In *2011 10th Roedunet International Conference (RoEduNet)*, pages 1–4. IEEE / Institute of Electrical and Electronics Engineers Incorporated, 2011. ISBN 978-1-4577-1233-3. doi:10.1109/RoEduNet.2011.5993718.

[158] Free Software Foundation, Inc. et. al. Copyleft and the GNU General Public License: A Comprehensive Tutorial and Guide: (Revision: a2d90d4b73ff), 2018. URL `https://k.copyleft.org/guide`.

[159] S. Bedanta, A. Barman, W. Kleemann, O. Petracic, and T. Seki. Magnetic Nanoparticles: A Subject for Both Fundamental Research and Applications. *Journal of Nanomaterials*, 2013:1–22, 2013. ISSN 1687-4110. doi:10.1155/2013/952540.

[160] P. Tartaj, M. P. Morales, T. González-Carreño, S. Veintemillas-Verdaguer, and C. J. Serna. Advances in magnetic nanoparticles for biotechnology applications. *Journal of Magnetism and Magnetic Materials*, 290-291:28–34, 2005. ISSN 0304-8853. doi:10.1016/j.jmmm.2004.11.155.

[161] T. J. Silva and W. H. Rippard. Developments in nano-oscillators

based upon spin-transfer point-contact devices. *Journal of Magnetism and Magnetic Materials*, 320(7):1260–1271, 2008. ISSN 0304-8853. doi:10.1016/j.jmmm.2007.12.022. URL `http://www.sciencedirect.com/science/article/pii/S0304885307010153`.

[162] Luc Piraux, Krystel Renard, Raphael Guillemet, Stefan Matéfi-Tempfli, Maria Matéfi-Tempfli, Vlad Andrei Antohe, Stéphane Fusil, Karim Bouzehouane, and Vincent Cros. Template-grown NiFe/Cu/NiFe nanowires for spin transfer devices. *Nano letters*, 7(9):2563–2567, 2007. doi:10.1021/nl070263s.

[163] Stuart Parkin and See-Hun Yang. Memory on the racetrack. *Nature nanotechnology*, 10(3):195–198, 2015. doi:10.1038/nnano.2015.41.

[164] Stuart S. P. Parkin, Masamitsu Hayashi, and Luc Thomas. Magnetic domain-wall racetrack memory. *Science*, 320(5873):190–194, 2008. ISSN 1095-9203. doi:10.1126/science.1145799.

[165] D. A. Allwood, G. Xiong, C. C. Faulkner, D. Atkinson, D. Petit, and R. P. Cowburn. Magnetic domain-wall logic. *Science*, 309(5741):1688–1692, 2005. ISSN 1095-9203. doi:10.1126/science.1108813.

[166] Niklas Romming, Christian Hanneken, Matthias Menzel, Jessica E. Bickel, Boris Wolter, Kirsten von Bergmann, André Kubetzka, and Roland Wiesendanger. Writing and deleting single magnetic skyrmions. *Science*, 341(6146): 636–639, 2013. ISSN 1095-9203. doi:10.1126/science.1240573.

[167] Jing Ma, Jiamian Hu, Zheng Li, and Ce-Wen Nan. Recent progress in multiferroic magnetoelectric composites: from bulk to thin films. *Advanced materials (Deerfield Beach, Fla.)*, 23(9):1062–1087, 2011. doi:10.1002/adma.201003636.

[168] M. Gibertini, M. Koperski, A. F. Morpurgo, and K. S. Novoselov. Magnetic 2D materials and heterostructures. *Nature nanotechnology*, 14(5):408–419, 2019. doi:10.1038/s41565-019-0438-6.

[169] L. Wang, I. Meric, P. Y. Huang, Q. Gao, Y. Gao, H. Tran, T. Taniguchi, K. Watanabe, L. M. Campos, D. A. Muller, J. Guo, P. Kim, J. Hone, K. L. Shepard, and C. R. Dean. One-dimensional electrical contact to a two-dimensional material. *Science*, 342(6158):614–617, 2013. ISSN 1095-9203. doi:10.1126/science.1244358.

[170] Elsa Lhotel, Ludovic D. C. Jaubert, and Peter C. W. Holdsworth. Fragmentation in Frustrated Magnets: A Review. *Journal of Low Temperature Physics*, 2020. ISSN 0022-2291. doi:10.1007/s10909-020-02521-3.

[171] Linus Pauling. The Structure and Entropy of Ice and of Other Crystals with Some Randomness of Atomic Arrangement. *Journal of the American Chemical Society*, 57(12):2680–2684, 1935. ISSN 0002-7863. doi:10.1021/ja01315a102.

[172] Steven T. Bramwell and Mark J. Harris. The history of spin ice. *Journal of physics. Condensed matter : an Institute of Physics journal*, 32(37):374010, 2020. doi:10.1088/1361-648X/ab8423.

[173] S. Lendinez and M. B. Jungfleisch. Magnetization dynamics in artificial spin ice. *Journal of physics. Condensed matter : an Institute of Physics journal*, 32(1):013001, 2020. doi:10.1088/1361-648X/ab3e78.

[174] Sandra H. Skjærvø, Christopher H. Marrows, Robert L. Stamps, and Laura J. Heyderman. Advances in artificial spin ice. *Nature Reviews Physics*, 2(1): 13–28, 2020. doi:10.1038/s42254-019-0118-3.

[175] Peter Fischer, Dédalo Sanz-Hernández, Robert Streubel, and Amalio Fernández-Pacheco. Launching a new dimension with 3D magnetic nanostructures. *APL Materials*, 8(1):010701, 2020. doi:10.1063/1.5134474.

[176] V. V. Kruglyak, S. O. Demokritov, and D. Grundler. Magnonics. *Journal of Physics D: Applied Physics*, 43(26):264001, 2010. ISSN 0022-3727. doi:10.1088/0022-3727/43/26/264001. URL `https://iopscience.iop.org/article/10.1088/0022-3727/43/26/264001#back-to-top-target`.

[177] C. Castelnovo, R. Moessner, and S. L. Sondhi. Magnetic monopoles in spin ice. *Nature*, 451(7174):42–45, 2008. ISSN 1476-4687. doi:10.1038/nature06433.

[178] S. T. Bramwell and M. J. Gingras. Spin ice state in frustrated magnetic pyrochlore materials. *Science*, 294(5546):1495–1501, 2001. ISSN 1095-9203. doi:10.1126/science.1064761.

[179] J. M. de Teresa and A. Fernández-Pacheco. Present and future applications of magnetic nanostructures grown by FEBID. *Applied Physics A*, 117(4):1645–1658, 2014. ISSN 0947-8396. doi:10.1007/s00339-014-8617-7.

[180] Harald Plank, Robert Winkler, Christian H. Schwalb, Johanna Hütner, Jason D. Fowlkes, Philip D. Rack, Ivo Utke, and Michael Huth. Focused Electron Beam-Based 3D Nanoprinting for Scanning Probe Microscopy: A Review. *Micromachines*, 11(1), 2019. ISSN 2072-666X. doi:10.3390/mi11010048.

[181] Michael Huth, Fabrizio Porrati, Christian Schwalb, Marcel Winhold, Roland Sachser, Maja Dukic, Jonathan Adams, and Georg Fantner. Focused electron

beam induced deposition: A perspective. *Beilstein journal of nanotechnology*, 3:597–619, 2012. doi:10.3762/bjnano.3.70.

[182] S. J. Randolph, J. D. Fowlkes, and P. D. Rack. Focused, Nanoscale Electron-Beam-Induced Deposition and Etching. *Critical Reviews in Solid State and Materials Sciences*, 31(3):55–89, 2006. ISSN 1040-8436. doi:10.1080/10408430600930438.

[183] W. F. van Dorp and C. W. Hagen. A critical literature review of focused electron beam induced deposition. *Journal of Applied Physics*, 104 (8):081301, 2008. ISSN 0021-8979. doi:10.1063/1.2977587. URL `https://aip.scitation.org/doi/10.1063/1.2977587`.

[184] Michael Huth, Lukas Keller, Harald Plank, and Robert Winkler. Kleingedrucktes mit großem Effekt. *Physik in unserer Zeit*, 51(2):64–71, 2020. ISSN 0031-9252. doi:10.1002/piuz.201901561.

[185] Amalio Fernández-Pacheco, Luka Skoric, José María de Teresa, Javier Pablo-Navarro, Michael Huth, and Oleksandr V. Dobrovolskiy. Writing 3D Nanomagnets Using Focused Electron Beams. *Materials (Basel, Switzerland)*, 13 (17):3774, 2020. ISSN 1996-1944. doi:10.3390/ma13173774. URL `https://www.mdpi.com/1996-1944/13/17/3774`.

[186] Jason D. Fowlkes, R. Winkler, Brett B. Lewis, A. Fernández-Pacheco, L. Skoric, D. Sanz-Hernández, Michael G. Stanford, Eva Mutunga, P. D. Rack, and H. Plank. High-Fidelity 3D-Nanoprinting via Focused Electron Beams: Computer-Aided Design (3BID). *ACS Applied Nano Materials*, 1(3):1028–1041, 2018. ISSN 2574-0970. doi:10.1021/acsanm.7b00342.

[187] F. Porrati, M. Pohlit, J. Müller, S. Barth, F. Biegger, C. Gspan, H. Plank, and M. Huth. Direct writing of CoFe alloy nanostructures by focused electron beam induced deposition from a heteronuclear precursor. *Nanotechnology*, 26(47):475701, 2015. doi:10.1088/0957-4484/26/47/475701. URL `https://iopscience.iop.org/article/10.1088/0957-4484/26/47/475701/meta`.

[188] Javier Pablo-Navarro, Dédalo Sanz-Hernández, César Magén, Amalio Fernández-Pacheco, and José María de Teresa. Tuning shape, composition and magnetization of 3D cobalt nanowires grown by focused electron beam induced deposition (FEBID). *Journal of Physics D: Applied Physics*, 50 (18):18LT01, 2017. ISSN 0022-3727. doi:10.1088/1361-6463/aa63b4. URL `https://iopscience.iop.org/article/10.1088/1361-6463/aa63b4`.

[189] Paul M. Weirich, Marcel Winhold, Christian H. Schwalb, and Michael Huth.

In situ growth optimization in focused electron-beam induced deposition. *Beilstein journal of nanotechnology*, 4:919–926, 2013. doi:10.3762/bjnano.4.103.

[190] R. Lavrijsen, R. Córdoba, F. J. Schoenaker, T. H. Ellis, B. Barcones, J. T. Kohlhepp, H. J. M. Swagten, B. Koopmans, J. M. de Teresa, C. Magén, M. R. Ibarra, P. Trompenaars, and J. J. L. Mulders. Fe:O:C grown by focused-electron-beam-induced deposition: magnetic and electric properties. *Nanotechnology*, 22(2):025302, 2011. doi:10.1088/0957-4484/22/2/025302. URL `https://iopscience.iop.org/article/10.1088/0957-4484/22/2/025302/meta`.

[191] A. Fernández-Pacheco, J. M. de Teresa, R. Córdoba, and M. R. Ibarra. Magnetotransport properties of high-quality cobalt nanowires grown by focused-electron-beam-induced deposition. *Journal of Physics D: Applied Physics*, 42(5):055005, 2009. ISSN 0022-3727. doi:10.1088/0022-3727/42/5/055005. URL `https://iopscience.iop.org/article/10.1088/0022-3727/42/5/055005/meta`.

[192] Lukas Keller, Mohanad K. I. Al Mamoori, Jonathan Pieper, Christian Gspan, Irina Stockem, Christian Schröder, Sven Barth, Robert Winkler, Harald Plank, Merlin Pohlit, Jens Müller, and Michael Huth. Direct-write of free-form building blocks for artificial magnetic 3D lattices. *Scientific reports*, 8(1):6160, 2018. doi:10.1038/s41598-018-24431-x.

[193] Merlin Pohlit, Irina Stockem, Fabrizio Porrati, Michael Huth, Christian Schröder, and Jens Müller. Experimental and theoretical investigation of the magnetization dynamics of an artificial square spin ice cluster. *Journal of Applied Physics*, 120(14):142103, 2016. ISSN 0021-8979. doi:10.1063/1.4961705.

[194] A. Candini, G. C. Gazzadi, A. Di Bona, M. Affronte, D. Ercolani, G. Biasiol, and L. Sorba. Hall nano-probes fabricated by focused ion beam. *Nanotechnology*, 17(9):2105–2109, 2006. ISSN 0957-4484. doi:10.1088/0957-4484/17/9/005. URL `https://iopscience.iop.org/article/10.1088/0957-4484/17/9/005`.

[195] Mihai Gabureac, Laurent Bernau, Ivo Utke, and Giovanni Boero. Granular Co-C nano-Hall sensors by focused-beam-induced deposition. *Nanotechnology*, 21(11):115503, 2010. doi:10.1088/0957-4484/21/11/115503. URL `https://iopscience.iop.org/article/10.1088/0957-4484/21/11/115503/meta`.

[196] Matthew Hunt, Mike Taverne, Joseph Askey, Andrew May, Arjen van den Berg, Ying-Lung Daniel Ho, John Rarity, and Sam Ladak. Harnessing Multi-Photon Absorption to Produce Three-Dimensional Magnetic Structures at

the Nanoscale. *Materials (Basel, Switzerland)*, 13(3), 2020. ISSN 1996-1944. doi:10.3390/ma13030761.

[197] Gwilym Williams, Matthew Hunt, Benedikt Boehm, Andrew May, Michael Taverne, Daniel Ho, Sean Giblin, Dan Read, John Rarity, Rolf Allenspach, and Sam Ladak. Two-photon lithography for 3D magnetic nanostructure fabrication: Nano Research, 11(2), 845-854. *Nano Research*, 11(2):845–854, 2017. ISSN 1998-0124. doi:10.1007/s12274-017-1694-0.

[198] Michael Huth, Fabrizio Porrati, Peter Gruszka, and Sven Barth. Temperature-Dependent Growth Characteristics of Nb- and CoFe-Based Nanostructures by Direct-Write Using Focused Electron Beam-Induced Deposition. *Micromachines*, 11(1), 2019. ISSN 2072-666X. doi:10.3390/mi11010028.

[199] Lukas Keller and Michael Huth. Pattern generation for direct-write three-dimensional nanoscale structures via focused electron beam induced deposition. *Beilstein journal of nanotechnology*, 9:2581–2598, 2018. doi:10.3762/bjnano.9.240.

[200] Mohanad K. I. Al Mamoori, Lukas Keller, Jonathan Pieper, Sven Barth, Robert Winkler, Harald Plank, Jens Müller, and Michael Huth. Magnetic Characterization of Direct-Write Free-Form Building Blocks for Artificial Magnetic 3D Lattices. *Materials (Basel, Switzerland)*, 11(2), 2018. ISSN 1996-1944. doi:10.3390/ma11020289.

[201] Mohanad Al Mamoori, Christian Schröder, Lukas Keller, Michael Huth, and Jens Müller. First-order reversal curves (FORCs) of nano-engineered 3D Co-Fe structures. *AIP Advances*, 10(1):015319, 2020. doi:10.1063/1.5129850.

[202] Mohanad K. I. Al Mamoori. *3D Nanomagnetic Architectures Investigated by micro-Hall Sensors*. Dissertation, TU Darmstadt, Germany, 2020.

[203] Silke Paschen and Qimiao Si. Quantum phases driven by strong correlations. *Nature Reviews Physics*, 3(1):9–26, 2021. doi:10.1038/s42254-020-00262-6.

[204] John Clarke. SQUIDs. *Scientific American*, 271(2):46–53, 1994. ISSN 00368733. URL `www.jstor.org/stable/24942801`.

[205] R. S. Popović. *Hall effect devices*. Series in sensors. Institute of Physics Publ, Bristol, 2. ed. edition, 2004. ISBN 0750308559. doi:10.1887/0750308559.

[206] T. M. Hengstmann, D. Grundler, Ch. Heyn, and D. Heitmann. Stray-field investigation on permalloy nanodisks. *Journal of Applied Physics*, 90(12): 6542–6544, 2001. ISSN 0021-8979. doi:10.1063/1.1413238.

[207] S. J. Bending and A. Oral. Hall effect in a highly inhomogeneous magnetic field distribution. *Journal of Applied Physics*, 81(8):3721–3725, 1997. ISSN 0021-8979. doi:10.1063/1.365494. URL `http://link.aip.org/link/?JAP/81/3721/1`.

[208] Morgan W. Mitchell and Silvana Palacios Alvarez. Colloquium : Quantum limits to the energy resolution of magnetic field sensors. *Reviews of Modern Physics*, 92(2), 2020. ISSN 0034-6861. doi:10.1103/RevModPhys.92.021001.

[209] Jan Dauber, Abhay A. Sagade, Martin Oellers, Kenji Watanabe, Takashi Taniguchi, Daniel Neumaier, and Christoph Stampfer. Ultra-sensitive Hall sensors based on graphene encapsulated in hexagonal boron nitride. *Applied Physics Letters*, 106(19):193501, 2015. ISSN 1077-3118. doi:10.1063/1.4919897.

[210] Jens Müller, Yongqing Li, Stephan von Molnár, Yuzo Ohno, and Hideo Ohno. Single-electron switching in AlxGa1−xAs/GaAs Hall devices. *Phys. Rev. B*, 74(12), 2006. ISSN 1095-3795. doi:10.1103/PhysRevB.74.125310.

[211] Yongqing Li. *Optimization of Hall Magnetometry and Single Magnetic Nanoparticle Measurements*. PhD thesis, The Florida State University - College of Art and Sciences, 2003.

[212] Yongqing Li, Cong Ren, Peng Xiong, Stephan von Molnár, Yuzo Ohno, and Hideo Ohno. Modulation of Noise in Submicron GaAs/AlGaAs Hall Devices by Gating. *Phys. Rev. Lett.*, 93(24):246602, 2004. doi:10.1103/PhysRevLett.93.246602. URL `http://link.aps.org/doi/10.1103/PhysRevLett.93.246602`.

[213] Rudolf Gross and Achim Marx. *Festkörperphysik*. Mathematik, Physik 10-2012. Oldenbourg, München, 2012. ISBN 978-3-486-71294-0. doi:10.1524/9783486714869. URL `http://www.oldenbourg-link.com/isbn/9783486712940`.

[214] B. Chenaud, A. Segovia-Mera, A. Delgard, N. Feltin, A. Hoffmann, F. Pascal, W. Zawadzki, D. Mailly, and C. Chaubet. Sensitivity and noise of micro-Hall magnetic sensors based on InGaAs quantum wells. *Journal of Applied Physics*, 119(2):024501, 2016. ISSN 0021-8979. doi:10.1063/1.4939288.

[215] Yongqing Li, Peng Xiong, Stephan von Molnár, Steffen Wirth, Yuzo Ohno, and Hideo Ohno. Hall magnetometry on a single iron nanoparticle. *Applied Physics Letters*, 80(24):4644, 2002. ISSN 0003-6951. doi:10.1063/1.1487921. URL `http://dx.doi.org/10.1063/1.1487921`.

[216] Mark D. McDonnell and Lawrence M. Ward. The benefits of noise in neural systems: bridging theory and experiment. *Nature reviews. Neuroscience*, 12 (7):415–426, 2011. doi:10.1038/nrn3061.

[217] J. P. Sethna, K. A. Dahmen, and C. R. Myers. Crackling noise. *Nature*, 410 (6825):242–250, 2001. ISSN 1476-4687. doi:10.1038/35065675.

[218] Rudolf Müller. *Rauschen: Zweite, überarbeitete und erweiterte Auflage*, volume 15 of *Halbleiter-Elektronik, Eine aktuelle Buchreihe für Studierende und Ingenieure*. Springer Berlin Heidelberg, Berlin, Heidelberg and s.l., 1990. ISBN 978-3-540-51145-8. doi:10.1007/978-3-642-61501-6.

[219] P. Welch. The use of fast Fourier transform for the estimation of power spectra: A method based on time averaging over short, modified periodograms. *IEEE Transactions on Audio and Electroacoustics*, 15(2):70–73, 1967. ISSN 0018-9278. doi:10.1109/TAU.1967.1161901.

[220] Jens Müller. Fluctuation spectroscopy: a new approach for studying low-dimensional molecular metals. *Chemphyschem : a European journal of chemical physics and physical chemistry*, 12(7):1222–1245, 2011. doi:10.1002/cphc.201000814.

[221] J. B. JOHNSON. Thermal Agitation of Electricity in Conductors. *Physical Review*, 32(1):97–109, 1928. ISSN 0031-899X. doi:10.1103/PhysRev.32.97.

[222] H. Nyquist. Thermal Agitation of Electric Charge in Conductors. *Physical Review*, 32(1):110–113, 1928. ISSN 0031-899X. doi:10.1103/PhysRev.32.110.

[223] Eugene T. Patronis, H. Marshak, C. A. Reynolds, V. L. Sailor, and F. J. Shore. Low–Temperature Thermal Noise Thermometer. *Review of Scientific Instruments*, 30(7):578–580, 1959. ISSN 1089-7623. doi:10.1063/1.1716687.

[224] D. Rothfuß, A. Reiser, A. Fleischmann, and C. Enss. Noise thermometry at ultra low temperatures. *Applied Physics Letters*, 103(5):052605, 2013. ISSN 1077-3118. doi:10.1063/1.4816760. URL `https://aip.scitation.org/doi/10.1063/1.4816760`.

[225] W. Schottky. Über spontane Stromschwankungen in verschiedenen Elektrizitätsleitern. *Annalen der Physik*, 362(23):541–567, 1918. ISSN 1521-3889. doi:10.1002/andp.19183622304.

[226] Carlo Beenakker and Christian Schönenberger. Quantum Shot Noise. *Physics Today*, 56(5):37–42, 2003. ISSN 0031-9228. doi:10.1063/1.1583532. URL `https://physicstoday.scitation.org/doi/10.1063/1.1583532`.

[227] Stefan Machlup. Noise in Semiconductors: Spectrum of a Two–Parameter Random Signal. *Journal of Applied Physics*, 25(3):341–343, 1954. ISSN 0021-8979. doi:10.1063/1.1721637.

[228] Marek Wolf. 1/f noise in the distribution of prime numbers. *Physica A: Statistical Mechanics and its Applications*, 241(3-4):493–499, 1997. ISSN 03784371. doi:10.1016/S0378-4371(97)00251-3.

[229] Toshimitsu Musha and Hideyo Higuchi. The 1/ f Fluctuation of a Traffic Current on an Expressway. *Japanese Journal of Applied Physics*, 15 (7):1271–1275, 1976. ISSN 1347-4065. doi:10.1143/JJAP.15.1271. URL `https://iopscience.iop.org/article/10.1143/JJAP.15.1271`.

[230] RICHARD F. VOSS and John Clarke. '1/fnoise' in music and speech. *Nature*, 258(5533):317–318, 1975. ISSN 1476-4687. doi:10.1038/258317a0.

[231] Tibor Grasser. *Noise in nanoscale semiconductor devices*. Springer International Publishing, Cham, 2020. ISBN 978-3-030-37499-0. doi:10.1007/978-3-030-37500-3.

[232] Alexander A. Balandin. Low-frequency 1/f noise in graphene devices. *Nature nanotechnology*, 8(8):549–555, 2013. doi:10.1038/nnano.2013.144.

[233] Kiyoo Itoh, Takayasu Sakurai, Fabrizio Bonani, and Giovanni Ghione. *Noise in Semiconductor Devices*, volume 7. Springer Berlin Heidelberg, Berlin, Heidelberg, 2001. ISBN 978-3-642-08586-4. doi:10.1007/978-3-662-04530-5.

[234] H. M. J. Vaes and T. G. M. Kleinpenning. Hall–effect noise in semiconductors. *Journal of Applied Physics*, 48(12):5131–5134, 1977. ISSN 0021-8979. doi:10.1063/1.323591.

[235] T. G. M. Kleinpenning. 1/ f noise in Hall effect: Fluctuations in mobility. *Journal of Applied Physics*, 51(6):3438, 1980. ISSN 0021-8979. doi:10.1063/1.328029.

[236] R. N. Jabdaraghi, D. S. Golubev, J. P. Pekola, and J. T. Peltonen. Noise of a superconducting magnetic flux sensor based on a proximity Josephson junction. *Sci. Rep.*, 7(1):8011, 2017. ISSN 2045-2322. doi:10.1038/s41598-017-08710-7.

[237] Brian T. Schaefer, Lei Wang, Alexander Jarjour, Kenji Watanabe, Takashi Taniguchi, Paul L. McEuen, and Katja C. Nowack. Magnetic field detection limits for ultraclean graphene Hall sensors. *Nature communications*, 11(1): 4163, 2020. doi:10.1038/s41467-020-18007-5.

[238] J. Muller, B. Korbitzer, A. Amyan, M. Pohlit, Y. Ohno, and H. Ohno. Noise spectroscopy studies of GaAs/AlGaAs hall devices for optimizing micro- and nano-scale magnetic measurements. In *2015 International Conference on Noise and Fluctuations (ICNF)*, pages 1–4, Piscataway, NJ, 2015. IEEE. ISBN 978-1-4673-8335-6. doi:10.1109/ICNF.2015.7288544.

[239] C. W. Hicks, L. Luan, K. A. Moler, E. Zeldov, and H. Shtrikman. Noise characteristics of 100nm scale GaAs/AlxGa1−xAs scanning Hall probes. *Applied Physics Letters*, 90(13):133512, 2007. ISSN 1077-3118. doi:10.1063/1.2717565.

[240] Jens Müller, Stephan von Molnár, Yuzo Ohno, and Hideo Ohno. Decomposition of 1/f Noise in AlxGa1−xAs/GaAs Hall Devices. *Physical Review Letters*, 96(18), 2006. ISSN 1079-7114. doi:10.1103/PhysRevLett.96.186601.

[241] Zoe Budrikis, K. L. Livesey, J. P. Morgan, J. Akerman, A. Stein, S. Langridge, C. H. Marrows, R. L. Stamps, and Z. Budrikis. Domain dynamics and fluctuations in artificial square ice at finite temperatures. *New Journal of Physics*, 14(3):035014, 2012. ISSN 1367-2630. doi:10.1088/1367-2630/14/3/035014.

[242] A. Ozbay, A. Gokce, T. Flanagan, R. A. Stearrett, E. R. Nowak, and C. Nordman. Low frequency magnetoresistive noise in spin-valve structures. *Applied Physics Letters*, 94(20):202506, 2009. ISSN 0003-6951. doi:10.1063/1.3139067.

[243] O. G. Shpyrko, E. D. Isaacs, J. M. Logan, Yejun Feng, G. Aeppli, R. Jaramillo, H. C. Kim, T. F. Rosenbaum, P. Zschack, M. Sprung, S. Narayanan, and A. R. Sandy. Direct measurement of antiferromagnetic domain fluctuations. *Nature*, 447(7140):68–71, 2007. ISSN 1476-4687. doi:10.1038/nature05776.

[244] O'Brien and Weissman. Statistical characterization of Barkhausen noise. *Physical review. E, Statistical physics, plasmas, fluids, and related interdisciplinary topics*, 50(5):3446–3452, 1994. ISSN 1063-651X. doi:10.1103/physreve.50.3446.

[245] Ritika Dusad, Franziska K. K. Kirschner, Jesse C. Hoke, Benjamin R. Roberts, Anna Eyal, Felix Flicker, Graeme M. Luke, Stephen J. Blundell, and J. C. Séamus Davis. Magnetic monopole noise. *Nature*, 571(7764):234–239, 2019. ISSN 1476-4687. doi:10.1038/s41586-019-1358-1.

[246] Alexey V. Klyuev, Mikhail I. Ryzhkin, and Arkady V. Yakimov. Statistics of Fluctuations of Magnetic Monopole Concentration in Spin Ice. *Fluctuation and Noise Letters*, 16(04):1750035, 2017. ISSN 0219-4775. doi:10.1142/S0219477517500353.

[247] S. M. Anton, J. S. Birenbaum, S. R. O'Kelley, V. Bolkhovsky, D. A. Braje, G. Fitch, M. Neeley, G. C. Hilton, H-M Cho, K. D. Irwin, F. C. Wellstood,

W. D. Oliver, A. Shnirman, and John Clarke. Magnetic flux noise in dc SQUIDs: temperature and geometry dependence. *Physical Review Letters*, 110(14):147002, 2013. ISSN 1079-7114. doi:10.1103/PhysRevLett.110.147002.

[248] M. J. Ferrari, Mark Johnson, Frederick C. Wellstood, John Clarke, P. A. Rosenthal, R. H. Hammond, and M. R. Beasley. Magnetic flux noise in thin–film rings of $YBa_2Cu_3O_{7-\delta}$. *Applied Physics Letters*, 53(8):695–697, 1988. ISSN 0003-6951. doi:10.1063/1.100643.

[249] E. Saitoh, K. Harii, H. Miyajima, and T. Yamaoka. Critical phenomena in magnetic vortex formation probed by noise spectroscopy. *Journal of Magnetism and Magnetic Materials*, 282:25–27, 2004. ISSN 0304-8853. doi:10.1016/j.jmmm.2004.04.007.

[250] T. Tsuboi, T. Hanaguri, and A. Maeda. Local Density Fluctuations of Moving Vortices in the Solid and Liquid Phases in Bi2Sr2CaCu2Oy. *Physical Review Letters*, 80(20):4550–4553, 1998. ISSN 1079-7114. doi:10.1103/PhysRevLett.80.4550.

[251] M. J. Ferrari, Mark Johnson, F. C. Wellstood, J. J. Kingston, T. J. Shaw, and John Clarke. Magnetic flux noise in copper oxide superconductors. *Journal of Low Temperature Physics*, 94(1-2):15–61, 1994. ISSN 0022-2291. doi:10.1007/BF00755416.

[252] G. Bertotti, G. Durin, and A. Magni. Scaling aspects of domain wall dynamics and Barkhausen effect in ferromagnetic materials. *Journal of Applied Physics*, 75(10):5490–5492, 1994. ISSN 0021-8979. doi:10.1063/1.355666.

[253] Bruno Alessandro, Cinzia Beatrice, Giorgio Bertotti, and Arianna Montorsi. Domain–wall dynamics and Barkhausen effect in metallic ferromagnetic materials. I. Theory. *Journal of Applied Physics*, 68(6):2901–2907, 1990. ISSN 0021-8979. doi:10.1063/1.346423.

[254] Bruno Alessandro, Cinzia Beatrice, Giorgio Bertotti, and Arianna Montorsi. Domain–wall dynamics and Barkhausen effect in metallic ferromagnetic materials. II. Experiments. *Journal of Applied Physics*, 68(6):2908–2915, 1990. ISSN 0021-8979. doi:10.1063/1.346424.

[255] B. Alessandro, C. Beatrice, G. Bertotti, and A. Montorsi. Phenomenology and interpretation of the Barkhausen effect in ferromagnetic materials (invited). *Journal of Applied Physics*, 64(10):5355–5360, 1988. ISSN 0021-8979. doi:10.1063/1.342370.

[256] G. Bertotti, F. Fiorillo, and M. P. Sassi. Barkhausen noise and domain struc-

ture dynamics in Si-Fe at different points of the magnetization curve. *Journal of Magnetism and Magnetic Materials*, 23(2):136–148, 1981. ISSN 0304-8853. doi:10.1016/0304-8853(81)90127-X.

[257] Adham Amayan. *Elektronische und magnetische Phasenseparation in* EuB_6*: Fluktuationsspektroskopie und nichtlinearer Transport.* Dissertation, Goethe Universität, Frankfurt am Main, 2013.

[258] Sarah Ottersbach. *Fluktuationsspektroskopie mittels schneller Datenerfassung und softwaregestützter Datenanalyse: Anwendung auf den Ladungsordnungsübergang in Nickelaten.* Master Thesis, Goethe Universität, Frankfurt am Main, 2013.

[259] Benedikt Hartmann. *Ladungsträgerdynamik am Mott-Übergang.* Dissertation, Goethe Universität, Frankfurt am Main, 2017.

[260] Jonathan Pieper. *Micro-Hall Magnetometry: Comparison of different measurement techniques for 3D Nanostructures.* Bachelor Thesis, Goethe Universität, Frankfurt am Main, 2017.

[261] Benjamin Kammerbauer. *Aufbau und Programmierung der Messmethode der dielektrischen Spektroskopie.* Bachelor Thesis, Goethe Universität, Frankfurt am Main, 2020.

[262] David Zielke. *Fluktuationsspektroskopie an quasi-zweidimensionalen molekularen Metallen.* Master Thesis, Goethe Universität, Frankfurt am Main, 2014.

[263] Pandoc. a universal document converter, 03.11.2020. URL `https://pandoc.org/`.

[264] Tim Thyzel. *Fluktuationsspektroskopie der ladungsgeordneten und glasartigen Phasen von* $\theta-(\mathrm{BEDT}-\mathrm{TTF})\_2\mathrm{TlZn}(\mathrm{SCN})_4$. Master Thesis, Goethe Universität, Frankfurt am Main, 2020.

[265] M. B. Weissman. What is a spin glass? A glimpse via mesoscopic noise. *Reviews of Modern Physics*, 65(3):829–839, 1993. ISSN 0034-6861. doi:10.1103/RevModPhys.65.829.

[266] Merlin Pohlit. *Micro-Hall-Magnetometrie: Streufeldmessungen an magnetischen Mikro- und Nanostrukturen.* Dissertation, Goethe Universität, Frankfurt am Main, 2017.

[267] Stefanie Scholz Christian Winkler. Datenvisualisierung mit Jupyter-

Notebooks, Teil 1. *iX*, 01/2021:56–61, 2020-12-16. URL https://www.heise.de/select/ix/2021/1/2026813023795667120.

[268] Stefanie Scholz Christian Winkler. Datenvisualisierung mit Jupyter-Notebooks, Teil 2. *iX*, 02/2021:130–133, 2021-01-20. URL https://www.heise.de/select/ix/2021/2/2034212181752486918.

[269] Paul F. Dubois. Testing Scientific Programs. *Computing in Science & Engineering*, 14(4):69–73, 2012. ISSN 1521-9615. doi:10.1109/MCSE.2012.84.

[270] Mohammed Shamsul Arefeen and Michael Schiller. Continuous Integration Using Gitlab. *Undergraduate Research in Natural and Clinical Science and Technology (URNCST) Journal*, 3(8):1–6, 2019. doi:10.26685/urncst.152.

[271] The Carpentries Incubator. Introduction to Open Data Science with R, 02.10.2020. URL https://carpentries-incubator.github.io/open-science-with-r/.

[272] The Carpentries Incubator. Introduction to Conda for (Data) Scientists, 04.11.2020. URL https://carpentries-incubator.github.io/introduction-to-conda-for-data-scientists/.

[273] The Carpentries Incubator. Reproducible Computational Environments Using Containers: Introduction to Docker, 21.10.2020. URL https://carpentries-incubator.github.io/docker-introduction/.

[274] Andreas Wiegenstein. Quelltext im Fokus: Werkzeuge zur automatischen Codeanalyse: Codescanner. *iX*, 11/2020:68ff., 2020-10-21. URL https://www.heise.de/select/ix/2020/11/1920610044330220665.

[275] Christian Rost, Thorsten Schifferdecker, and Christian Schneemann. Gut sortiert: Hilfe bei Microservices: Kubernetes mit Support. *iX*, 12/2019:50ff., 2019-11-20. URL https://www.heise.de/select/ix/2019/12/1908709553513964488.

[276] Patrick Münch Christoph Puppe. Sicher am laufenden Band: Docker-Container und -Hosts überwachen. *iX*, 5/2017:78–81, 2017-04-19. URL https://www.heise.de/select/ix/2017/5/1492959394520870.

[277] Monya Baker. Scientific computing: Code alert. *Nature*, 541(7638):563–565, 2017. ISSN 1476-4687. doi:10.1038/nj7638-563a.

[278] Makoto Isoda, Hiroki Nakano, and Tôru Sakai. Frustration-Induced Magnetic Properties of the Spin-1/2 Heisenberg Antiferromagnet on the Cairo Pentagon

Lattice. *Journal of the Physical Society of Japan*, 83(8):084710, 2014. ISSN 0031-9015. doi:10.7566/JPSJ.83.084710.

[279] Maryam Akbari Lalehdashti. *Herstellung und physikalische Charakterisierung von GaAs/AlGaAs Mikro-Hall-Sensoren*. Master Thesis, Goethe Universität, Frankfurt am Main, 2016.

[280] Giorgio Bertotti. *Hysteresis in magnetism: For physicists, materials scientists and engineers*. Academic Press series in electromagnetism. Acad. Press, San Diego, Calif., digitaler nachdr edition, 2008. ISBN 0-12-093270-9.

[281] Robin E. Bell. Gravity Gradiometry. *Scientific American*, 278(6):74–79, 1998. ISSN 0036-8733. doi:10.1038/scientificamerican0698-74.

[282] Zhu Diao, E. R. Nowak, Gen Feng, and J. M. D. Coey. Magnetic noise in structured hard magnets. *Physical Review Letters*, 104(4):047202, 2010. ISSN 1079-7114. doi:10.1103/PhysRevLett.104.047202.

[283] Merlin Pohlit, Fabrizio Porrati, Michael Huth, Yuzo Ohno, Hideo Ohno, and Jens Müller. Nanocluster building blocks of artificial square spin ice: Stray-field studies of thermal dynamics. *Journal of Applied Physics*, 117(17):17C746, 2015. ISSN 0021-8979. doi:10.1063/1.4917497. URL `http://dx.doi.org/10.1063/1.4917497`.

[284] Berit Körbitzer. *Fluktuationsspektroskopie an zweidimensionalen Elektronensystemen in GaAs/AlGaAs-Heterostrukturen*. Master Thesis, Goethe Universität, Frankfurt am Main, 2012.

[285] Y. G. Cornelissens and F. M. Peeters. Response function of a Hall magnetosensor in the diffusive regime. *Journal of Applied Physics*, 92(4):2006–2012, 2002. ISSN 0021-8979. doi:10.1063/1.1487909. URL `http://link.aip.org/link/?JAP/92/2006/1`.

[286] C.W.J. Beenakker and H. van Houten. Quantum Transport in Semiconductor Nanostructures. *Solid State Physics*, 44:1–228, 1991. doi:10.1016/S0081-1947(08)60091-0.

[287] GitHub. The State of the Octoverse, 16.12.2020. URL `https://octoverse.github.com/`.

[288] GitLab. Bug Bounty Program, 25.01.2021. URL `https://hackerone.com/gitlab`.

[289] Stefan Walluhn. DevOps Disasters 3.1, 16.01.2021. URL https://media.ccc.de/v/rc3-49321-devops_disasters_3_1#t=202.

[290] Olivia von Westernhagen. Schubladen für Schwachstellen: Das CVE-System im Überblick. *heise Online*, 2020-03-11. URL https://www.heise.de/hintergrund/Schubladen-fuer-Schwachstellen-Das-CVE-System-im-Ueberblick-4940478.html.

[291] Jan Bundesmann. Freihändig: Open-Source-Tools fürs Konfigurationsmanagement. *iX*, 1/2021:96–103, 2020-12-16. URL https://www.heise.de/select/ix/2021/1/2004408295806428798.

[292] Ute Roos. Security Scorecards sollen Sicherheit von Open-Source-Paketen einschätzen. *heise Online*, 2020-11-11. URL https://www.heise.de/news/Security-Scorecards-sollen-Sicherheit-von-Open-Source-Paketen-einschaetzen-4 html.

[293] Andreas Kurtz. Von niedrig bis kritisch: Schwachstellenbewertung mit CVSS. *heise Online*, 2021-01-25. URL https://www.heise.de/hintergrund/Von-niedrig-bis-kritisch-Schwachstellenbewertung-mit-CVSS-5031983.html.

[294] GitHub. coreinfrastructure/best-practices-badge, 01.09.2020. URL https://github.com/coreinfrastructure/best-practices-badge/blob/master/doc/background.md.

[295] The tox automation project. standardize testing in Python, 23.7.2020. URL https://tox.readthedocs.io/en/latest/.

[296] Silke Hahn. Versionsverwaltung: GitLab führt neues Abomodell ein. *heise Online*, 2021-01-27. URL https://heise.de/-5038294.

[297] Alexander Rahn Eldar Sultanow. Visualisieren großer Datenmengen mit Python und Blender. *iX*, 01/2021:62–67, 2020-12-16. URL https://www.heise.de/select/ix/2021/1/2030813060694921498.

[298] albi3ro. Metals, Magnets, and Miscellaneous Materials, 27.08.2020. URL http://albi3ro.github.io/M4/.

[299] John T. Wolohan. *Mastering Large Datasets with Python : Parallelize and Distribute Your Python Code*. Manning Publications Company, 2020. ISBN 9781617296239.

[300] Kieran Joseph Healy. *Data visualization: A practical introduction*. Princeton University Press, Princeton and Oxford, 2019. ISBN 9780691181615.

[301] Nico Axtmann. In Betrieb: Vom Modell zum produktiven Einsatz. *iX special*, 15:60–63, 2020. URL https://www.heise.de/select/ix-special/archiv/2020/15/seite-60.

[302] Alyssa Goodman, Josh Peek, Alberto Accomazzi, Chris Beaumont, Christine L. Borgman, How-Huan Hope Chen, Merce Crosas, Christopher Erdmann, August Muench, Alberto Pepe, and Curtis Wong. The "Paper" of the Future. doi:10.22541/au.148769949.92783646.

[303] GitLab. GitLab for education, 01.02.2021. URL https://about.gitlab.com/solutions/education/.

[304] Universität Bremen. Dienste/Gitlab, 03.02.2021. URL http://www.informatik.uni-bremen.de/t/Dienste/Gitlab.

[305] Johannes Gutenberg-Unversität Mainz. Zentrum für Datenverarbeitung: Gitlab, 03.02.2021. URL https://www.zdv.uni-mainz.de/gitlab/.

[306] Martin-Luther-Universität Halle-Wittenberg. Gitlab Institut für Informatik der MLU, 03.02.2021. URL https://gitlab.informatik.uni-halle.de/users/sign_in.

[307] Gottfried Wilhelm Leibniz Universität Hannover. GitLab, 03.02.2021. URL https://www.luis.uni-hannover.de/de/services/anwendungen/gitlab/.

[308] Universität Leipzig. URZ - GitLab, 03.02.2021. URL https://www.urz.uni-leipzig.de/fue/sc/gitlab/.

[309] University of Reading. Academic Computing Team: GitLab (Git repository), 23.04.2020. URL https://research.reading.ac.uk/act/knowledgebase/gitlab-git-repository/.

[310] Ulm University. Faculty GitLab, 2020. URL https://www.uni-ulm.de/en/in/sgi/services/faculty-gitlab/.

[311] Julius-Maximilians-Universität Würzburg. GitLab, 09.11.2020. URL https://wiki.informatik.uni-wuerzburg.de/gitlab.html.

[312] Bert Besser. Gut verwaltet: Data Version Control im Team mit Open-

Source-Werkzeugen. *iX special*, 15:48–53, 2020. URL https://www.heise.de/select/ix-special/archiv/2020/15/seite-48.

[313] Rohit Batra. Accurate machine learning in materials science facilitated by using diverse data sources. *Nature*, 589(7843):524–525, 2021. ISSN 1476-4687. doi:10.1038/d41586-020-03259-4.

[314] Eliska Greplova. Solving optimization tasks in condensed matter. *Nature Machine Intelligence*, 2(10):557–558, 2020. doi:10.1038/s42256-020-00240-8.

[315] Hans-Christian Ruiz Euler, Marcus N. Boon, Jochem T. Wildeboer, Bram van de Ven, Tao Chen, Hajo Broersma, Peter A. Bobbert, and Wilfred G. van der Wiel. A deep-learning approach to realizing functionality in nanoelectronic devices. *Nature nanotechnology*, 15(12):992–998, 2020. doi:10.1038/s41565-020-00779-y.

[316] Yashar Kiarashinejad, Sajjad Abdollahramezani, and Ali Adibi. Deep learning approach based on dimensionality reduction for designing electromagnetic nanostructures. *npj Computational Materials*, 6(1), 2020. doi:10.1038/s41524-020-0276-y.

[317] Jonathan D. Cryer and Kung-sik Chan. *Time series analysis: With applications in R*. Springer texts in statistics. Springer, New York, second edition edition, 2010. ISBN 9780387759593.

[318] Björn Schelter, Jens Timmer, and Matthias Winterhalder. *Handbook of time series analysis: Recent theoretical developments and applications*. Wiley-VCH, Weinheim, 2006. ISBN 9783527609970. doi:10.1002/9783527609970.

[319] Anna Lorenz. Blick in die Kristallkugel: Methoden für Zeitreihenvorhersagen. *iX special*, 15:100–107, 2020. URL https://www.heise.de/select/ix-special/archiv/2020/15/seite-100.

[320] Löning, Markus and Bagnall, Anthony and Ganesh, Sajaysurya and Kazakov, Viktor and Lines, Jason and Király, Franz J. sktime: A Unified Interface for Machine Learning with Time Series. 24.07.2020. URL http://learningsys.org/neurips19/assets/papers/sktime_ml_systems_neurips2019.pdf.

[321] Alexander Alexandrov, Konstantinos Benidis, Michael Bohlke-Schneider, Valentin Flunkert, Jan Gasthaus, Tim Januschowski, Danielle C. Maddix, Syama Rangapuram, David Salinas, Jasper Schulz, Lorenzo Stella, Ali Caner Türkmen, and Yuyang Wang. GluonTS: Probabilistic Time Series Models in Python. URL https://arxiv.org/pdf/1906.05264.

[322] Gianluca Bontempi, Souhaib Ben Taieb, and Yann-Aël Le Borgne. Machine Learning Strategies for Time Series Forecasting. In Marie-Aude Aufaure and Esteban Zimányi, editors, *Business Intelligence*, volume 138 of *Lecture Notes in Business Information Processing*, pages 62–77. Springer Berlin Heidelberg, Berlin/Heidelberg, 2013. ISBN 978-3-642-36317-7. doi:10.1007/978-3-642-36318-4_3.

[323] Ali Taylan Cemgil, David Barber, and Silvia Chiappa, editors. *Bayesian time series models*. Cambridge University Press, Cambridge, 2011. ISBN 9780521196765. doi:10.1017/CBO9780511984679.

[324] Nesreen K. Ahmed, Amir F. Atiya, Neamat El Gayar, and Hisham El-Shishiny. An Empirical Comparison of Machine Learning Models for Time Series Forecasting. *Econometric Reviews*, 29(5-6):594–621, 2010. ISSN 0747-4938. doi:10.1080/07474938.2010.481556.

[325] Christopher M. Bishop. *Pattern recognition and machine learning*. Springer, New York, NY, 9. (corrected at 8.th printing) edition, 2009. ISBN 0-387-31073-8.

[326] Choose a License. GNU General Public License v3.0, 30.10.2020. URL `https://choosealicense.com/licenses/gpl-3.0/`.

[327] P. Barrett, J. Hunter, J. T. Miller, J.-C. Hsu, and P. Greenfield. matplotlib – A Portable Python Plotting Package. *Astronomical Data Analysis Software and Systems XIV ASP Conference Series, Vol. 347, Proceedings of the Conference held 24-27 October, 2004 in Pasadena, California, USA.*, page 91, 2005. URL `http://adsabs.harvard.edu/full/2005ASPC..347...91B`.

[328] Choose a License. BSD 3-Clause "New" or "Revised" License, 30.10.2020. URL `https://choosealicense.com/licenses/bsd-3-clause/`.

[329] Choose a License. Apache License 2.0, 11.11.2020. URL `https://choosealicense.com/licenses/apache-2.0/`.

[330] Anaconda Inc. The World's Most Popular Data Science Platform, 13.11.2020. URL `https://www.anaconda.com/`.

[331] John Garrett. garrettj403/SciencePlots (v1.0.6), 2020.

[332] Creative Commons. Attribution 4.0 International License (CC BY 4.0), 07.11.2020. URL `https://creativecommons.org/licenses/by/4.0/`.

[333] Open Source Initiative. The MIT License, 26.7.2020. URL https://opensource.org/licenses/MIT.

[334] Creative Commons. Attribution-NonCommercial-ShareAlike 3.0 Unported (CC BY-NC-SA 3.0), 19.06.2021. URL https://creativecommons.org/licenses/by-nc-sa/3.0/.

[335] Open Knowledge Open Definition Group. The Open Definition, 03.10.2020. URL https://opendefinition.org/.

[336] Python. Documentation: Glossary: Version 3.9.1, 2020. URL https://docs.python.org/3/glossary.html.

APPENDIX A

SUPPLEMENTAL INFORMATION

A.1 Data, Code and Documentation

Supplemental information about data and code are available in the Lab–Book. The Lab–Book's structure is shown in Figure A.2. Source code and data documentation is mainly written by experimenters and coders to be helpful for their future–self and later adapted for reviewers, spending a reasonable amount of time.

Figure A.1 shows the file structure of the Lab–Book's repository. Documentation and analysis scripts are in subfolders `docs/` and `docs/notebooks/`, respectively. The original OneNote Notebook exports are available inside the `source/onenote/` folder. Test cases to recreate the plots from Chapter 4 can be found in `ana`'s test folder[1].

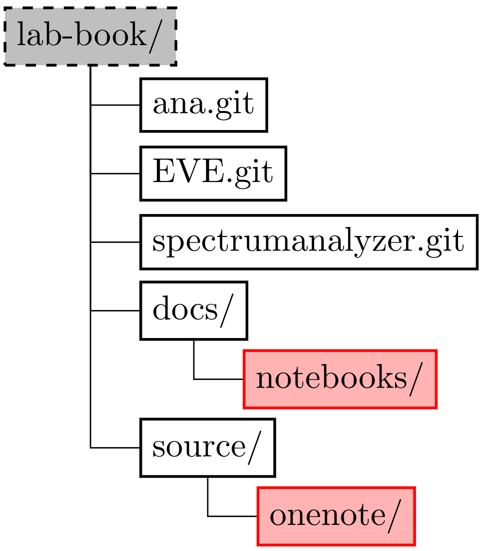


Figure A.1.: **Project folder organization.**

[1] `ana.git/tests/ana/visualize/test_master_plots.py`

Note

Data, code, and documentation is available for reviewers. The supplemental information is provided through a self–maintained server on the internet, and available for the review period:

`https://master.ody5.de`

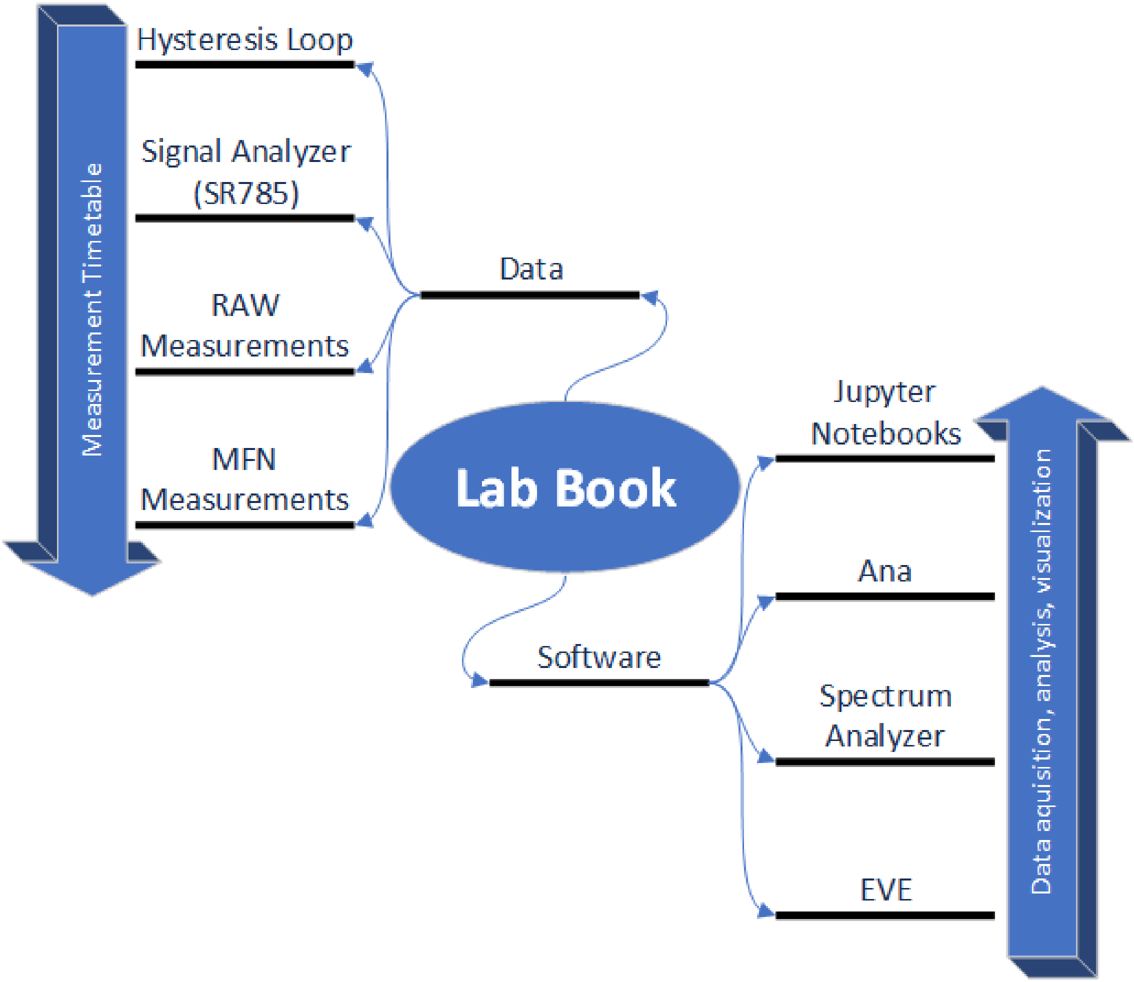


Figure A.2.: **Structure of supplemental information in the Lab Book.**

APPENDIX B

LEGAL NOTICES

Figure B.1 depicts an overview of common open–source licenses. All documentation regarding the data, including created graphs, pictures, diagrams, and context, can be licensed using Creative Commons licenses [155]. Source code for computer programs and scripts should take advantage of the number of open–source licenses [62] to protect the work from unwanted infringements or other legal problems.

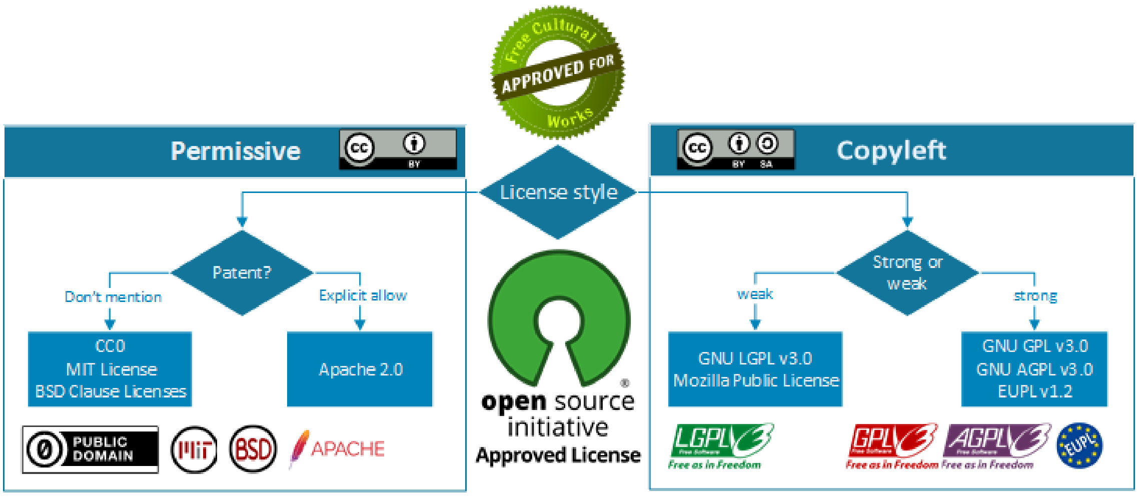


Figure B.1.: **Open Source Licenses**. A selection of popular open source licenses ordered from most permissive (left) to restricted (right).

The programmed software builds on open–source modules. The usage of these modules requires compliance with the agreed license. One such requirement involves for derived works to include the copyright owner and license of the software. Figure B.2 shows the software dependencies of EVE and the spectrumanalyzer and the

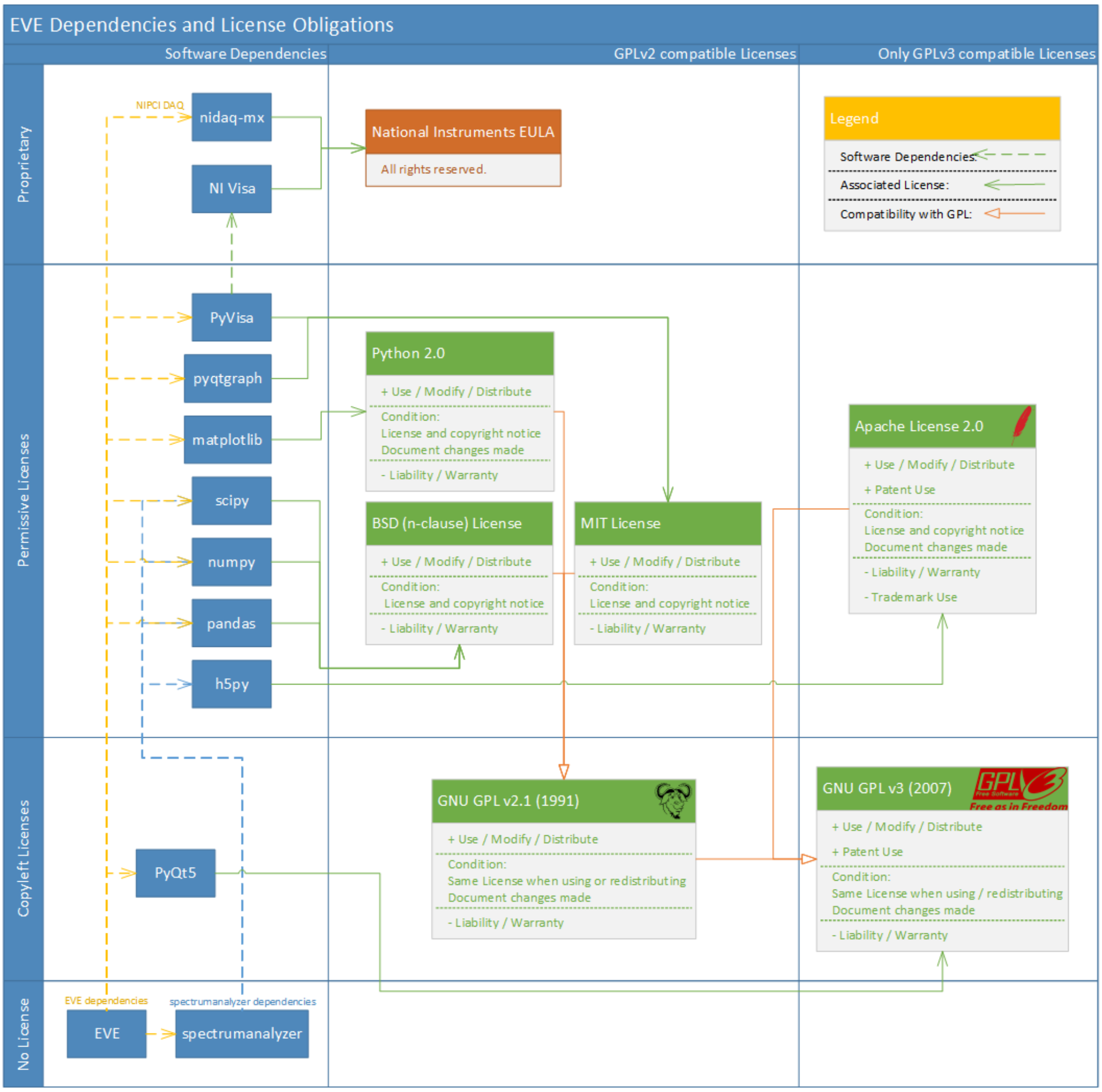


Figure B.2.: **Python dependencies of EVE and spectrumanalyzer and resulting license compliances.**

corresponding licenses. Table B.1 and B.2 shows the tools that are involved in the creation of this thesis.

Name	Version	Copyright	License
Python	3.8	Python Software Foundation	PSF Python 3.9.0
EVE	ebdb829	AGM (B. Hartmann et al.)	
spectrumanalyzer	3e30865	AGM (A. Amayan et al.)	
ana	7923164	Jonathan Pieper	GNU GPLv3 [326]
matplotlib	3.3.3	Paul Barrett [327]	matplotlib
seaborn	0.11.1	Michael L. Waskom	BSD (3–Clause) [328]
numpy	1.19.5	Stefan van der Walt [138]	BSD (3–Clause) [328]
SciPy	1.6.0	Pauli Virtanen [137]	BSD (3–Clause) [328]
Pandas	1.2.0	NumFOCUS	BSD (3–Clause) [328]
h5py	3.1.0	The HDF Group	Apache 2.0 [329]
pyqtgraph	0.12.2	Luke Campagnola	MIT [156]
PyVisa	1.11.3	PyVISA Authors	MIT [156]
NI-Visa	20.0	National Instruments	NI EULA
NI-DAQmx	20.0	National Instruments	NI EULA
nidaqmx	0.5.7	National Instruments	MIT [156]
PyQT5	5.15.2	Riverbank Computing	GNU GPLv3 [326]

Table B.1.: Software and licenses connected to research study.

B.1 License and Copyright notices

Thesis Content and LaTeX Code

Python Data Analysis Framework (Ana) and Jupyter Notebooks

[1] `https://www.latextemplates.com/template/the-legrand-orange-book`
[2] legrand.mathias@gmail.com
[3] vel@latextemplates.com

Name	Version	Copyright	License
Ubuntu	18.04	Cannonical Ltd.	GNU GPLv3 [326]
GitLab	13.x.x	GitLab B.V.	MIT [156]
Gitlab–Runner	13.x.x	GitLab B.V.	MIT [156]
Docker Desktop	20.10.8	Docker	Apache 2.0 [329]
Jupyterlab Image	763e2154	Jonathan Pieper	MIT [156]
Jupyter Core	4.7.0	Project Jupyter	BSD (3–Clause) [328]
Jupyter Lab	3.0.5	Project Jupyter	BSD (3–Clause) [328]
Anaconda		Anaconda Inc. [330]	BSD (3–Clause) [328]
Pandoc	2.10.1	[263]	BSD (3–Clause) [328]
Sphinx	3.4.3	[142]	BSD (3–Clause) [328]
IPython	7.19.0	Fernando Pérez [140]	BSD (3–Clause) [328]
coverage	5.3.1	Ned Batchelder	Apache 2.0 [329]
SciencePlots	ac8d772	John Garrett [331]	MIT [156]

Table B.2.: Additional software and licensed used in continuous analysis.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see `https://www.gnu.org/licenses/`.« [326]

Lab–Book

The Lab–Book as a compendium of all its containing code, data, documentation, and examples is a database by definition of the German »Urheberrecht« [154, §87a]. Everyone is welcome to redistribute it and/or modify it under the terms of the Creative Commons Attribution 4.0 International (**CC BY** 4.0 [332]) license.

APPENDIX C

OPEN–SOURCE LICENSES

The following open–source licenses are part of the most popular and permissive licenses available.

C.1 MIT License

Permission is hereby granted, free of charge, to any person obtaining a copy of this software and associated documentation files (the "Software"), to deal in the Software without restriction, including without limitation the rights to use, copy, modify, merge, publish, distribute, sublicense, and/or sell copies of the Software, and to permit persons to whom the Software is furnished to do so, subject to the following conditions:

The above copyright notice and this permission notice shall be included in all copies or substantial portions of the Software.

THE SOFTWARE IS PROVIDED "AS IS", WITHOUT WARRANTY OF ANY KIND, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE AND NONINFRINGEMENT. IN NO EVENT SHALL THE AUTHORS OR COPYRIGHT HOLDERS BE LIABLE FOR ANY CLAIM, DAMAGES OR OTHER LIABILITY, WHETHER IN AN ACTION OF CONTRACT, TORT OR

OTHERWISE, ARISING FROM, OUT OF OR IN CONNECTION WITH THE SOFTWARE OR THE USE OR OTHER DEALINGS IN THE SOFTWARE.

C.2 BSD (3–Clause) License

Redistribution and use in source and binary forms, with or without modification, are permitted provided that the following conditions are met:

1. Redistributions of source code must retain the above copyright notice, this list of conditions and the following disclaimer.
2. Redistributions in binary form must reproduce the above copyright notice, this list of conditions and the following disclaimer in the documentation and/or other materials provided with the distribution.
3. Neither the name of the copyright holder nor the names of its contributors may be used to endorse or promote products derived from this software without specific prior written permission.

THIS SOFTWARE IS PROVIDED BY THE COPYRIGHT HOLDERS AND CONTRIBUTORS "AS IS" AND ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE ARE DISCLAIMED. IN NO EVENT SHALL THE COPYRIGHT HOLDER OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

C.3 GNU GENERAL PUBLIC LICENSE

Version 3, 29 June 2007

Copyright © 2007 Free Software Foundation, Inc. `https://fsf.org/`

Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed.

C.3.1 Preamble

The GNU General Public License is a free, copyleft license for software and other kinds of works.

The licenses for most software and other practical works are designed to take away your freedom to share and change the works. By contrast, the GNU General Public License is intended to guarantee your freedom to share and change all versions of a program–to make sure it remains free software for all its users. We, the Free Software Foundation, use the GNU General Public License for most of our software; it applies also to any other work released this way by its authors. You can apply it to your programs, too.

When we speak of free software, we are referring to freedom, not price. Our General Public Licenses are designed to make sure that you have the freedom to distribute copies of free software (and charge for them if you wish), that you receive source code or can get it if you want it, that you can change the software or use pieces of it in new free programs, and that you know you can do these things.

To protect your rights, we need to prevent others from denying you these rights or asking you to surrender the rights. Therefore, you have certain responsibilities if you distribute copies of the software, or if you modify it: responsibilities to respect the freedom of others.

For example, if you distribute copies of such a program, whether gratis or for a fee, you must pass on to the recipients the same freedoms that you received. You must make sure that they, too, receive or can get the source code. And you must show them these terms so they know their rights.

Developers that use the GNU GPL protect your rights with two steps: (1) assert copyright on the software, and (2) offer you this License giving you legal permission to copy, distribute and/or modify it.

For the developers' and authors' protection, the GPL clearly explains that there is no warranty for this free software. For both users' and authors' sake, the GPL requires that modified versions be marked as changed, so that their problems will not be attributed erroneously to authors of previous versions.

Some devices are designed to deny users access to install or run modified versions of the software inside them, although the manufacturer can do so. This is fundamentally incompatible with the aim of protecting users' freedom to change the software. The systematic pattern of such abuse occurs in the area of products for individuals to use, which is precisely where it is most unacceptable. Therefore, we have designed this version of the GPL to prohibit the practice for those products. If such problems arise substantially in other domains, we stand ready to extend this provision to those domains in future versions of the GPL, as needed to protect the freedom of users.

Finally, every program is threatened constantly by software patents. States should not allow patents to restrict development and use of software on general-purpose computers, but in those that do, we wish to avoid the special danger that patents applied to a free program could make it effectively proprietary. To prevent this, the GPL assures that patents cannot be used to render the program non-free.

The precise terms and conditions for copying, distribution and modification follow.

C.3.2 TERMS AND CONDITIONS

0. Definitions.

"This License" refers to version 3 of the GNU General Public License.

"Copyright" also means copyright-like laws that apply to other kinds of works, such as semiconductor masks.

"The Program" refers to any copyrightable work licensed under this License. Each licensee is addressed as "you". "Licensees" and "recipients" may be individuals or organizations.

To "modify" a work means to copy from or adapt all or part of the work in a fashion requiring copyright permission, other than the making of an exact copy. The resulting work is called a "modified version" of the earlier work or a work "based on" the earlier work.

A "covered work" means either the unmodified Program or a work based on the Program.

To "propagate" a work means to do anything with it that, without permission, would make you directly or secondarily liable for infringement under applicable copyright law, except executing it on a computer or modifying a private copy. Propagation includes copying, distribution (with or without modification), making available to the public, and in some countries other activities as well.

To "convey" a work means any kind of propagation that enables other parties to make or receive copies. Mere interaction with a user through a computer network, with no transfer of a copy, is not conveying.

An interactive user interface displays “Appropriate Legal Notices” to the extent that it includes a convenient and prominently visible feature that (1) displays an appropriate copyright notice, and (2) tells the user that there is no warranty for the work (except to the extent that warranties are provided), that licensees may convey the work under this License, and how to view a copy of this License. If the interface presents a list of user commands or options, such as a menu, a prominent item in the list meets this criterion.

1. Source Code.

The “source code” for a work means the preferred form of the work for making modifications to it. “Object code” means any non-source form of a work.

A “Standard Interface” means an interface that either is an official standard defined by a recognized standards body, or, in the case of interfaces specified for a particular programming language, one that is widely used among developers working in that language.

The “System Libraries” of an executable work include anything, other than the work as a whole, that (a) is included in the normal form of packaging a Major Component, but which is not part of that Major Component, and (b) serves only to enable use of the work with that Major Component, or to implement a Standard Interface for which an implementation is available to the public in source code form. A “Major Component”, in this context, means a major essential component (kernel, window system, and so on) of the specific operating system (if any) on which the executable work runs, or a compiler used to produce the work, or an object code interpreter used to run it.

The “Corresponding Source” for a work in object code form means all the source code needed to generate, install, and (for an executable work) run the object code and to modify the work, including scripts to control those activities. However, it does not include the work’s System Libraries, or general-purpose tools or generally available free programs which are used unmodified in performing those activities but which are not part of the work. For example, Corresponding Source includes interface definition files associated with source files for the work, and the source code for shared libraries and dynamically linked subprograms that the work is specifically designed to require, such as by intimate data communication or control flow between those subprograms and other parts of the work.

The Corresponding Source need not include anything that users can regenerate automatically from other parts of the Corresponding Source.

The Corresponding Source for a work in source code form is that same work.

2. Basic Permissions.

All rights granted under this License are granted for the term of copyright on the Program, and are irrevocable provided the stated conditions are met. This License explicitly affirms your unlimited permission to run the unmodified Program. The output from running a covered work is covered by this License only if the output, given its content, constitutes a covered work. This License acknowledges your rights of fair use or other equivalent, as provided by copyright law.

You may make, run and propagate covered works that you do not convey, without conditions so long as your license otherwise remains in force. You may convey covered works to others for the sole purpose of having them make modifications exclusively for you, or provide you with facilities for running those works, provided that you comply with the terms of this License in conveying all material for which you do not control copyright. Those thus making or running the covered works for you must do so exclusively on your behalf, under your direction and control, on terms that prohibit them from making any copies of your copyrighted material outside their relationship with you.

Conveying under any other circumstances is permitted solely under the conditions stated below. Sublicensing is not allowed; section 10 makes it unnecessary.

3. Protecting Users' Legal Rights From Anti-Circumvention Law.

No covered work shall be deemed part of an effective technological measure under any applicable law fulfilling obligations under article 11 of the WIPO copyright treaty adopted on 20 December 1996, or similar laws prohibiting or restricting circumvention of such measures.

When you convey a covered work, you waive any legal power to forbid circumvention of technological measures to the extent such circumvention is effected by exercising rights under this License with respect to the covered work, and you disclaim any intention to limit operation or modification of the work as a means of enforcing, against the work's users, your or third parties' legal rights to forbid circumvention of technological measures.

4. Conveying Verbatim Copies.

You may convey verbatim copies of the Program's source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy an appropriate copyright notice; keep intact all notices stating that this License and any non-permissive terms added in accord with section 7 apply to the code; keep intact all notices of the absence of any warranty; and give all recipients a copy of this License along with the Program.

You may charge any price or no price for each copy that you convey, and you may offer support or warranty protection for a fee.

5. Conveying Modified Source Versions.

You may convey a work based on the Program, or the modifications to produce it from the Program, in the form of source code under the terms of section 4, provided that you also meet all of these conditions:

1. The work must carry prominent notices stating that you modified it, and giving a relevant date.
2. The work must carry prominent notices stating that it is released under this License and any conditions added under section 7. This requirement modifies the requirement in section 4 to "keep intact all notices".
3. You must license the entire work, as a whole, under this License to anyone who comes into possession of a copy. This License will therefore apply, along with any applicable section 7 additional terms, to the whole of the work, and all its parts, regardless of how they are packaged. This License gives no permission to license the work in any other way, but it does not invalidate such permission if you have separately received it.
4. If the work has interactive user interfaces, each must display Appropriate Legal Notices; however, if the Program has interactive interfaces that do not display Appropriate Legal Notices, your work need not make them do so.

A compilation of a covered work with other separate and independent works, which are not by their nature extensions of the covered work, and which are not combined with it such as to form a larger program, in or on a volume of a storage or distribution medium, is called an "aggregate" if the compilation and its resulting copyright are not used to limit the access or legal rights of the compilation's users beyond what the individual works permit. Inclusion of a covered work in an aggregate does not cause this License to apply to the other parts of the aggregate.

6. Conveying Non-Source Forms.

You may convey a covered work in object code form under the terms of sections 4 and 5, provided that you also convey the machine-readable Corresponding Source under the terms of this License, in one of these ways:

1. Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by the Corresponding Source fixed on a durable physical medium customarily used for software interchange.
2. Convey the object code in, or embodied in, a physical product (including a physical distribution medium), accompanied by a written offer, valid for at least three years and valid for as long as you offer spare parts or customer support for that product model, to give anyone who possesses the object code either (1) a copy of the Corresponding Source for all the software in the product that is covered by this License, on a durable physical medium customarily used for software interchange, for a price no more than your reasonable cost of physically performing this conveying of source, or (2) access to copy the Corresponding Source from a network server at no charge.
3. Convey individual copies of the object code with a copy of the written offer to provide the Corresponding Source. This alternative is allowed only occasionally and noncommercially, and only if you received the object code with such an offer, in accord with subsection 6b.
4. Convey the object code by offering access from a designated place (gratis or for a charge), and offer equivalent access to the Corresponding Source in the same way through the same place at no further charge. You need not require recipients to copy the Corresponding Source along with the object code. If the place to copy the object code is a network server, the Corresponding Source may be on a different server (operated by you or a third party) that supports equivalent copying facilities, provided you maintain clear directions next to the object code saying where to find the Corresponding Source. Regardless of what server hosts the Corresponding Source, you remain obligated to ensure that it is available for as long as needed to satisfy these requirements.
5. Convey the object code using peer-to-peer transmission, provided you inform other peers where the object code and Corresponding Source of the work are being offered to the general public at no charge under subsection 6d.

A separable portion of the object code, whose source code is excluded from the Corresponding Source as a System Library, need not be included in conveying the object code work.

A "User Product" is either (1) a "consumer product", which means any tangible personal property which is normally used for personal, family, or household purposes, or (2) anything designed or sold for incorporation into a dwelling. In determining whether a product is a consumer product, doubtful cases shall be resolved in favor of coverage. For a particular product received by a particular user, "normally used" refers to a typical or common use of that class of product, regardless of the status of the particular user or of the way in which the particular user actually uses, or expects or is expected to use, the product. A product is a consumer product regardless of whether the product has substantial commercial, industrial or non-consumer uses, unless such uses represent the only significant mode of use of the product.

"Installation Information" for a User Product means any methods, procedures, authorization keys, or other information required to install and execute modified versions of a covered work in that User Product from a modified version of its Corresponding Source. The information must suffice to ensure that the continued functioning of the modified object code is in no case prevented or interfered with solely because modification has been made.

If you convey an object code work under this section in, or with, or specifically for use in, a User Product, and the conveying occurs as part of a transaction in which the right of possession and use of the User Product is transferred to the recipient in perpetuity or for a fixed term (regardless of how the transaction is characterized), the Corresponding Source conveyed under this section must be accompanied by the Installation Information. But this requirement does not apply if neither you nor any third party retains the ability to install modified object code on the User Product (for example, the work has been installed in ROM).

The requirement to provide Installation Information does not include a requirement to continue to provide support service, warranty, or updates for a work that has been modified or installed by the recipient, or for the User Product in which it has been modified or installed. Access to a network may be denied when the modification itself materially and adversely affects the operation of the network or violates the rules and protocols for communication across the network.

Corresponding Source conveyed, and Installation Information provided, in accord with this section must be in a format that is publicly documented (and with an implementation available to the public in source code form), and must require no special password or key for unpacking, reading or copying.

7. Additional Terms.

"Additional permissions" are terms that supplement the terms of this License by making exceptions from one or more of its conditions. Additional permissions that are applicable to the entire Program shall be treated as though they were included in this License, to the extent that they are valid under applicable law. If additional permissions apply only to part of the Program, that part may be used separately under those permissions, but the entire Program remains governed by this License without regard to the additional permissions.

When you convey a copy of a covered work, you may at your option remove any additional permissions from that copy, or from any part of it. (Additional permissions may be written to require their own removal in certain cases when you modify the work.) You may place additional permissions on material, added by you to a covered work, for which you have or can give appropriate copyright permission.

Notwithstanding any other provision of this License, for material you add to a covered work, you may (if authorized by the copyright holders of that material) supplement the terms of this License with terms:

1. Disclaiming warranty or limiting liability differently from the terms of sections 15 and 16 of this License; or
2. Requiring preservation of specified reasonable legal notices or author attributions in that material or in the Appropriate Legal Notices displayed by works containing it; or
3. Prohibiting misrepresentation of the origin of that material, or requiring that modified versions of such material be marked in reasonable ways as different from the original version; or
4. Limiting the use for publicity purposes of names of licensors or authors of the material; or
5. Declining to grant rights under trademark law for use of some trade names, trademarks, or service marks; or
6. Requiring indemnification of licensors and authors of that material by anyone who conveys the material (or modified versions of it) with contractual assumptions of liability to the recipient, for any liability that these contractual assumptions directly impose on those licensors and authors.

All other non-permissive additional terms are considered "further restrictions" within the meaning of section 10. If the Program as you received it, or any part of it, contains a notice stating that it is governed by this License along with a term that is a further restriction, you may remove that term. If a license document contains a further restriction but permits relicensing or conveying under this License, you may add to a covered work material governed by the terms of that license document, provided that the further restriction does not survive such relicensing or conveying.

If you add terms to a covered work in accord with this section, you must place, in the relevant source files, a statement of the additional terms that apply to those files, or a notice indicating where to find the applicable terms.

Additional terms, permissive or non-permissive, may be stated in the form of a separately written license, or stated as exceptions; the above requirements apply either way.

8. Termination.

You may not propagate or modify a covered work except as expressly provided under this License. Any attempt otherwise to propagate or modify it is void, and will automatically terminate your rights under this License (including any patent licenses granted under the third paragraph of section 11).

However, if you cease all violation of this License, then your license from a particular copyright holder is reinstated (a) provisionally, unless and until the copyright holder explicitly and finally terminates your license, and (b) permanently, if the copyright holder fails to notify you of the violation by some reasonable means prior to 60 days after the cessation.

Moreover, your license from a particular copyright holder is reinstated permanently if the copyright holder notifies you of the violation by some reasonable means, this is the first time you have received notice of violation of this License (for any work) from that copyright holder, and you cure the violation prior to 30 days after your receipt of the notice.

Termination of your rights under this section does not terminate the licenses of parties who have received copies or rights from you under this License. If your rights have been terminated and not permanently reinstated, you do not qualify to receive new licenses for the same material under section 10.

9. Acceptance Not Required for Having Copies.

You are not required to accept this License in order to receive or run a copy of the Program. Ancillary propagation of a covered work occurring solely as a consequence of using peer-to-peer transmission to receive a copy likewise does not require acceptance. However, nothing other than this License grants you permission to propagate or modify any covered work. These actions infringe copyright if you do not accept this License. Therefore, by modifying or propagating a covered work, you indicate your acceptance of this License to do so.

10. Automatic Licensing of Downstream Recipients.

Each time you convey a covered work, the recipient automatically receives a license from the original licensors, to run, modify and propagate that work, subject to this License. You are not responsible for enforcing compliance by third parties with this License.

An “entity transaction” is a transaction transferring control of an organization, or substantially all assets of one, or subdividing an organization, or merging organizations. If propagation of a covered work results from an entity transaction, each party to that transaction who receives a copy of the work also receives whatever licenses to the work the party’s predecessor in interest had or could give under the previous paragraph, plus a right to possession of the Corresponding Source of the work from the predecessor in interest, if the predecessor has it or can get it with reasonable efforts.

You may not impose any further restrictions on the exercise of the rights granted or affirmed under this License. For example, you may not impose a license fee, royalty, or other charge for exercise of rights granted under this License, and you may not initiate litigation (including a cross-claim or counterclaim in a lawsuit) alleging that any patent claim is infringed by making, using, selling, offering for sale, or importing the Program or any portion of it.

11. Patents.

A “contributor” is a copyright holder who authorizes use under this License of the Program or a work on which the Program is based. The work thus licensed is called the contributor’s “contributor version”.

A contributor’s “essential patent claims” are all patent claims owned or controlled by the contributor, whether already acquired or hereafter acquired, that would be infringed by some manner, permitted by this License, of making, using, or selling its contributor version, but do not include claims that would be infringed only as a consequence of further modification of the contributor version. For purposes of this definition, “control” includes the right to grant patent sublicenses in a manner consistent with the requirements of this License.

Each contributor grants you a non-exclusive, worldwide, royalty-free patent license under the contributor’s essential patent claims, to make, use, sell, offer for sale, import and otherwise run, modify and propagate the contents of its contributor version.

In the following three paragraphs, a “patent license” is any express agreement or commitment, however denominated, not to enforce a patent (such as an express permission to practice a patent or covenant not to sue for patent infringement). To “grant” such a patent license to a party means to make such an agreement or commitment not to enforce a patent against the party.

If you convey a covered work, knowingly relying on a patent license, and the Corresponding Source of the work is not available for anyone to copy, free of charge and under the terms of this License, through a publicly available network server or other readily accessible means, then you must either (1) cause the Corresponding Source to be so available, or (2) arrange to deprive yourself of the benefit of the patent license for this particular work, or (3) arrange, in a manner consistent with the requirements of this License, to extend the patent license to downstream recipients. “Knowingly relying” means you have actual knowledge that, but for the patent license, your conveying the covered work in a country, or your recipient’s use of the covered work in a country, would infringe one or more identifiable patents in that country that you have reason to believe are valid.

If, pursuant to or in connection with a single transaction or arrangement, you convey, or propagate by procuring conveyance of, a covered work, and grant a patent license to some of the parties receiving the covered work authorizing them to use, propagate, modify or convey a specific copy of the covered work, then the patent license you grant is automatically extended to all recipients of the covered work and works based on it.

A patent license is "discriminatory" if it does not include within the scope of its coverage, prohibits the exercise of, or is conditioned on the non-exercise of one or more of the rights that are specifically granted under this License. You may not convey a covered work if you are a party to an arrangement with a third party that is in the business of distributing software, under which you make payment to the third party based on the extent of your activity of conveying the work, and under which the third party grants, to any of the parties who would receive the covered work from you, a discriminatory patent license (a) in connection with copies of the covered work conveyed by you (or copies made from those copies), or (b) primarily for and in connection with specific products or compilations that contain the covered work, unless you entered into that arrangement, or that patent license was granted, prior to 28 March 2007.

Nothing in this License shall be construed as excluding or limiting any implied license or other defenses to infringement that may otherwise be available to you under applicable patent law.

12. No Surrender of Others' Freedom.

If conditions are imposed on you (whether by court order, agreement or otherwise) that contradict the conditions of this License, they do not excuse you from the conditions of this License. If you cannot convey a covered work so as to satisfy simultaneously your obligations under this License and any other pertinent obligations, then as a consequence you may not convey it at all. For example, if you agree to terms that obligate you to collect a royalty for further conveying from those to whom you convey the Program, the only way you could satisfy both those terms and this License would be to refrain entirely from conveying the Program.

13. Use with the GNU Affero General Public License.

Notwithstanding any other provision of this License, you have permission to link or combine any covered work with a work licensed under version 3 of the GNU Affero General Public License into a single combined work, and to convey the resulting work. The terms of this License will continue to apply to the part which is the covered work, but the special requirements of the GNU Affero General Public License, section 13, concerning interaction through a network will apply to the combination as such.

14. Revised Versions of this License.

The Free Software Foundation may publish revised and/or new versions of the GNU General Public License from time to time. Such new versions will be similar in spirit to the present version, but may differ in detail to address new problems or concerns.

Each version is given a distinguishing version number. If the Program specifies that a certain numbered version of the GNU General Public License "or any later version" applies to it, you have the option of following the terms and conditions either of that numbered version or of any later version published by the Free Software Foundation. If the Program does not specify a version number of the GNU General Public License, you may choose any version ever published by the Free Software Foundation.

If the Program specifies that a proxy can decide which future versions of the GNU General Public License can be used, that proxy's public statement of acceptance of a version permanently authorizes you to choose that version for the Program.

Later license versions may give you additional or different permissions. However, no additional obligations are imposed on any author or copyright holder as a result of your choosing to follow a later version.

15. Disclaimer of Warranty.

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

16. Limitation of Liability.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY

WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

17. Interpretation of Sections 15 and 16.

If the disclaimer of warranty and limitation of liability provided above cannot be given local legal effect according to their terms, reviewing courts shall apply local law that most closely approximates an absolute waiver of all civil liability in connection with the Program, unless a warranty or assumption of liability accompanies a copy of the Program in return for a fee.

END OF TERMS AND CONDITIONS

C.3.3 How to Apply These Terms to Your New Programs

If you develop a new program, and you want it to be of the greatest possible use to the public, the best way to achieve this is to make it free software which everyone can redistribute and change under these terms.

To do so, attach the following notices to the program. It is safest to attach them to the start of each source file to most effectively state the exclusion of warranty; and each file should have at least the "copyright" line and a pointer to where the full notice is found.

<one line to give the program's name and a brief idea of what it does.> Copyright (C) <year> <name of author>

This program is free software: you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation, either version 3 of the License, or (at your option) any later version.

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

You should have received a copy of the GNU General Public License along with this program. If not, see https://www.gnu.org/licenses/.

Also add information on how to contact you by electronic and paper mail.

If the program does terminal interaction, make it output a short notice like this when it starts in an interactive mode:

<program> Copyright (C) <year> <name of author> This program comes with ABSOLUTELY NO WARRANTY; for details type ‘show w’. This is free software, and you are welcome to redistribute it under certain conditions; type ‘show c’ for details.

The hypothetical commands ‘show w’ and ‘show c’ should show the appropriate parts of the General Public License. Of course, your program’s commands might be different; for a GUI interface, you would use an “about box”.

You should also get your employer (if you work as a programmer) or school, if any, to sign a “copyright disclaimer” for the program, if necessary. For more information on this, and how to apply and follow the GNU GPL, see https://www.gnu.org/licenses/.

The GNU General Public License does not permit incorporating your program into proprietary programs. If your program is a subroutine library, you may consider it more useful to permit linking proprietary applications with the library. If this is what you want to do, use the GNU Lesser General Public License instead of this License. But first, please read https://www.gnu.org/licenses/why-not-lgpl.html.

LIST OF FIGURES

NOMENCLATURE

2DEG two-dimensional electron gas, page 26

API application programming interface, page 12

CI Continuous Integration, page 9

CI/CD Continuous Integration / Continuous Development, page 4

csv comma-separated-value, page 10

DAQ data acquisition, page 9

EULA end user license agreement, page V

FAIR findability, accessibility, interoperability, and reusability, page 3

FEBID focused electron beam induced deposition, page 25

FFT Fast Fourier Transform, page 15

free software »The users have the freedom to run, copy, distribute, study, change and improve the software.« [158], page 2

GNU recursive acronym for »GNU's Not Unix«, page 5

GPIB General Purpose Interface Bus (IEEE–488), page 13

GPL General Public License, page 5

GUI graphical user interface, page 10

IDE integrated development environment, page 18

IVC inner vacuum chamber, page 29

KDE kernel density estimation, page 44

NI National Instruments, page 12

open »Open means anyone can freely access, use, modify, and share for any purpose (subject, at most, to requirements that preserve provenance and openness).« [335], page 2

PCI Peripheral Component Interconnect, page 12

PID proportional-integral-derivative, page 29

PSD power spectral density, page 12

Python class »A template for creating user–defined objects. Class definitions normally contain method definitions which operate on instances of the class.« [336], page 15

Python generator »A function which returns a generator iterator.« [336], page 15

Python iterator »An object representing a stream of data. Repeated calls to the iterator's `__next__()` method [...] return successive items in the stream.« [336], page 15

Python module »An object that serves as an organizational unit of Python code.« [336], page 12

Python package »A Python module which can contain submodules or recursively, subpackages.« [336], page 10

Regex A regular expression defines a search pattern., page 14

SQUID Superconducting Quantum Interference Device, page 6

UML unified modeling language, page 17

INDEX

Symbols

A

D

E

F

G

L

M

S

T

APPENDIX D
ACKNOWLEDGEMENTS

This thesis exemplifies my educational journey towards becoming a scientist. I could only venture this journey and acquire the necessary skills through the unimaginable support and brain–inspired guidance of various people on my path towards a master's degree. They paved the way for me to skilfully scrutinize scientific challenges while encouraging me to stay ever–curious and spurring me to forever improve myself. I am eternally grateful for this defining experience. The following lines attempt to express this profound gratitude.

First and foremost, I would like to enunciate my insurmountable gratitude to my supervisor, Prof. Dr. Jens Müller. I consider myself exceptionally fortunate to have enjoyed his everlasting patience, compassion, support, trust, and wisdom, as well as an extraordinary degree of freedom in conducting my studies. He meticulously elucidates even most complex issues and his outstanding human qualities and shrewd academic acumen zealously permeate his entire working group — creating an familiar, supportive, and constructive atmosphere. This atmosphere contributes immensely to the personal and professional development of the working group's members. Hence, it has been an invaluable privilege to perform my studies as a member of his working group.

Another source of vigorous inspiration was the collaboration with Mohanad Al Mamoori. His ever–helpful comments and wide–spread support immeasurably contributed to my maturation. I am utmostly grateful for his consistent availability and the number of unremitting and stimulating discussions throughout this thesis. I am delighted to have worked with Dr. Martin Lonsky, Tatjana Thomas, and Merlin Mitschek, who warm–heartednessly welcomed all my questions. Furthermore, I am cordially thankful to all other colleagues who supported me over the past.

Last but not least, I would like to express my sincere gratitude to Prof. Dr. Michael Huth. He created the nanostructures studied in this thesis, and, thereby, making this project possible in the first place. Complementary, I wish to thank Lucas Keller and Fabrizio Porrati for the collaboration in the growth of the nanostructures. Additionally, I am thankful to Prof. Dr. Christian Schröder (Bielefeld University of Applied Sciences) for his helpful collaboration.

Personally, I would like to extend my unreserved gratitude to my brother, Patrick Pieper. For his immeasurable help in my whole life, I am ardently thankful. I am incredibly grateful to my mother, Bärbel Pieper, my fathers, Wolfgang Kaiser and Achim Kessler, and my grandparents, Horst Kaiser and Margot Kaiser, for their unconditional support and love, enabling my academic journey in the first place.

ERKLÄRUNG

Hiermit erkläre ich, dass ich die Arbeit selbstständig und ohne Benutzung anderer als der angegebenen Quellen und Hilfsmittel verfasst habe. Alle Stellen der Arbeit, die wörtlich oder sinngemäß aus Veröffentlichungen oder aus anderen fremden Texten entnommen wurden, sind von mir als solche kenntlich gemacht worden. Ferner erkläre ich, dass die Arbeit nicht – auch nicht auszugsweise – für eine andere Prüfung verwendet wurde.

Frankfurt am Main, den 9. Februar 2021

Jonathan Pieper