ALMA MATER STUDIORUM
UNIVERSITÀ DI BOLOGNA

DOTTORATO DI RICERCA IN

MATEMATICA

Ciclo 37

Settore Concorsuale: 01/A5 - ANALISI NUMERICA

Settore Scientifico Disciplinare: MAT/08 - ANALISI NUMERICA

MATHEMATICAL MODELS AND NUMERICAL METHODS FOR ENVIRONMENTAL APPLICATIONS OF FAST FIELD CYCLING NUCLEAR MAGNETIC RESONANCE

Presentata da: Giovanni Vito Spinelli

Coordinatore Dottorato

Giovanni Mongardi

Supervisore

Fabiana Zama

Co-supervisore

Leonardo Brizi

Esame finale anno 2025

ABSTRACT

Fast Field-Cycling (FFC) Nuclear Magnetic Resonance (NMR) relaxometry is a powerful, non-destructive technique used to investigate molecular dynamics and structures across a wide range of systems, including environmental, biological, and food-related applications. By operating at low magnetic field intensities, FFC-NMR enables the exploration of slow molecular dynamics and offers detailed insights into molecular motion across diverse timescales within a single experiment. Despite its broad applicability, accurately identifying parameters from NMR Dispersion (NMRD) profiles remains a significant computational challenge.

This thesis introduces and evaluates advanced inverse methods based on regularization strategies and machine learning approaches to enhance the analysis of NMRD profiles. The *model-free* framework is explored, representing the NMRD profile R_1 as a linear combination of Lorentzian functions. To address the ill-conditioned nature of the parameter identification problem, three regularization-based methodologies are reviewed and validated: (1) a locally adaptive L_2 regularization method (*MF-UPen*), (2) an L_1-penalized approach (*MF-L1*), and (3) a hybrid method combining local L_2 and global L_1 penalties (*MF-MUPen*). These algorithms are further enhanced with automated regularization parameter selection using Balancing and Uniform Penalty principles, improving robustness and reproducibility.

Additionally, a constrained L_1-regularized non-linear least squares framework is proposed for modeling the parameter identification problem in the presence of the quadrupolar interaction, i.e., the quadrupolar relaxation enhancement (QRE) effect, due to electric interactions between nuclei with spins greater than 1/2 (e.g., ^{14}N) and the electric fields nearby. Building on prior methodologies, this approach decomposes the relaxation profiles into contributions associated with $^1H - {}^1H$ dipole-dipole interaction and $^1H - {}^{14}N$ quadrupolar one. The regularization parameter is iteratively computed via the Balancing Principle, while model parameters are optimized using a non-linear Gauss-Seidel algorithm. Several tests on both synthetic and real datasets validate the convergence properties and effectiveness of this approach. Additionally, the developed MATLAB tool based on this method is freely available for further research applications.

Finally, this thesis proposes a novel machine-learning framework based on the philosophy of the Plug-and-Play (PnP) technique to address the QRE phenomenon observed in FFC-NMR. A pre-trained feed-forward neural network incorporated into a coordinate descent optimization algorithm is employed to extract quadrupolar parameters and fit NMRD profiles. The network's custom loss function combines L_1 loss with accuracy in predicting quadrupolar components, enabling precise parameter extraction. Experimental validation against traditional optimization inverse methods highlights the framework's accuracy and potential for processing large datasets efficiently, particularly in industry applications.

This work advances the computational toolkit for FFC-NMR relaxometry, offering robust algorithms and machine-learning solutions that enhance the understanding of molecular dynamics across diverse systems.

Keywords: Nuclear Magnetic Resonance, Fast Field-Cycling, Relaxometry, Molecular Dynamics, Regularization Strategies, Inverse Problems, Quadrupole Relaxation Enhancement, Plug-And-Play, Machine Learning.

Contents

Introduction

This PhD thesis is developed under the project *Dottorati PON - Bando 2021 - Cycle 37 (XXXVII) - Action IV.5 - Doctorates on Green topics* - DOT1303154-2 supported by the Italian Ministry of Education and Merit, focusing on Innovation and Green topics. The National Operational Program (PON-green) aims to provide funds for research activities regarding green transition, ecosystem preservation, and reduction of climate change impacts. This project holds a constrained topic, i.e., *Mathematical Models and Numerical Methods for Environmental Application of Fast Field Cycling Nuclear Magnetic Resonance.*

Fast Field-Cycling (FFC) Nuclear Magnetic Resonance (NMR) relaxometry is a powerful, non-destructive magnetic resonance technique designed to explore slow molecular dynamics, accessible only at extremely low magnetic field strengths. Unlike standard NMR relaxation experiments, which are limited to a fixed and relatively large magnetic field determining the resonance frequency of the investigated molecules, FFC-NMR relaxometry enables relaxation studies across a vast frequency range (approximately 1 kHz to 40 MHz). This is achieved by varying the strength of the applied magnetic field to measure the longitudinal relaxation rate, R_1, forming Nuclear Magnetic Resonance Dispersion (NMRD) profiles. These profiles provide insights into molecular motion over a wide range of timescales (from milliseconds to picoseconds) within a single experiment.

Frequency-dependent relaxation studies have exceptional potential to reveal the timescales of molecular motion and the underlying mechanisms driving these motions. When the proper constraints are fulfilled [1, 2], spin relaxation can be described as a linear combination of spectral density functions, which are Fourier transforms of the time correlation functions. These functions capture the motional frequencies and their intensities in the correlation function. However, complex spin dynamical interactions, such as Quadrupole Relaxation Enhancement (QRE), can arise from intramolecular magnetic dipolar coupling with quadrupole nuclei (e.g., ^{14}N, with spin $S \geq 1$) [3, 4]. For these systems, QRE appears as local maxima or peaks in the R_1 profiles, with peak positions and shapes dependent on quadrupole parameters determined by the electric field gradient tensor at the ^{14}N position. Subtle changes in the electronic structure around ^{14}N can thus significantly affect the QRE signature, making it a sensitive fingerprint of molecular arrangements. This sensitivity has been exploited in diverse applications, including environmental science [5], ionic liquids [6], proteins [4], and food systems [2, 7].

Despite its broad utility, the adoption of FFC-NMR relaxometry faces challenges, including the complexity of data analysis and the absence of a unified computational framework for automating the interpretation of NMRD profiles. Specifically, for instance, selecting the correct approach to analyze data is highly dependent on the specific characteristics of the sample and the acquisition.

In the literature, there exist different software tools to address the analysis of NMRD profiles. For example, one of the most used in this research area, and widely well-known is Fitteia®. This software is an open-access web service (available at `http://fitteia.org`) that provides a user-friendly package for model fitting and data analysis [8]. This software, beyond the intuitiveness, incorporates the accumulated knowledge from the large number of model-fitting problems addressed by the authors, resulting in a powerful tool, especially in the NMR analysis data, which generally requires performing model fits of multi-dimensional

data sets and multi-dimensional model parameters space [8, 9, 10, 11].

Other interesting available software tools specifically for NMR analysis data, besides the more general ones (e.g., OriginPro [12], or QtiPlot [13]), are collected in a very interesting web service called NMRbox [14], where, depending on the problem, it is possible to find already developed packages and software to perform the analysis.

Although numerous software solutions have been developed to address the problem, a comprehensive framework remains lacking.

Applications of FFC-NMR relaxometry are numerous and extend to food science, material science, environmental studies, and biological research. In material science, for example, FFC-NMR has been applied to monitor molecular dynamics and subtle structural changes in macromolecular systems, providing a link between functional properties and molecular motion [15].

Regarding food science, for instance, this technique has proven valuable for quality control and authentication, such as distinguishing between authentic and adulterated cheeses like Pecorino and Parmigiano Reggiano [16, 17]. The non-destructive nature of FFC-NMR further allows continuous monitoring of food properties without altering the sample, enabling real-time quality assessments [18].

Similarly, in environmental research, FFC-NMR is widely used to study soil organic matter, contaminant interactions, and the porous structures of rocks. These studies contribute to a deeper understanding of carbon cycling, contaminant transport, and resource management [19, 20, 21, 22].

Despite these advancements, the adoption of FFC-NMR remains limited due to the sophisticated instrumentation and the need for specialized expertise in NMR and materials physics. Although artificial intelligence (AI) techniques have gained attraction in magnetic resonance imaging, their application to FFC-NMR is still emerging. Some AI-driven methods, such as convolutional neural networks (CNNs) for food quality assessment [23] and deep neural networks for reconstructing sparsely sampled spectra in NMR spectroscopy [24], suggest promising directions for future developments in this field.

This PhD work addresses the challenges of data analysis and the lack of a standardized computational framework for automating the interpretation of NMRD profiles and it relies on the Model-Free (MF) approach [25, 26] which represents the R_1 profile as a linear combination of Lorentzian functions. To analyze the data and identify the MF parameters, the problem is reformulated as an ill-conditioned least-squares problem that requires robust regularization techniques to ensure stable solutions [27, 28]. One critical aspect of this process is the selection of appropriate regularization parameters, which is essential for balancing prior information with data accuracy. Building on this foundation, after an in-depth discussion of the physical background and the Model-Free approach, the first part focuses on the discussion of inverse problems, regularization techniques, and the automatic computation of the regularization parameter.

The second part of this thesis, which illustrates the contribution of the PhD work, aims to explore several strategies consisting of both classical regularization techniques for ill-conditioned inverse problems, and hybrid techniques, based on AI approaches. In particular, regularization techniques, such as those based on L_1 or L_2 penalties, can be enhanced by incorporating machine learning models to infer meaningful priors or parameters from experimental datasets. Conversely, learning-based approaches can benefit from improved stability and interpretability when paired with regularization strategies rooted in physical modeling. For instance, hybrid frameworks that embed neural networks within traditional optimization schemes can simultaneously exploit the expressiveness of data-driven models and the theoretical rigor of regularized formulations. The proposed strategies demonstrate state-of-the-art performance, providing both computational efficiency and improved interpretability, which are crucial for advancing the analysis and inversion of FFC-NMR relaxometry data.

Application of NMR in Environmental and Food Research

The FFC-NMR technique plays a critical role in diverse areas of environmental research. Principal issues can be categorized into two main areas: molecular structure and molecular interactions [19].

Considering the first area, the NMR can be used to unravel the structure of soil organic matter (SOM). In environmental science, FFC-NMR aids in analyzing SOM and its response to climate change and agricultural practices. SOM was thought to consist mainly of humic substances cross-linked to form a unique chemical category. The use of NMR demonstrated that SOM is extremely complex. The alkaline extractable component of soil is dominated by plant and microbial biopolymers at various states of decay [20, 21]. Understanding SOM composition as a mixture of microbial and plant residues at different stages of degradation requires the use of specific NMR techniques such as FFC to study soil aggregate structure, humification processes, fertility, and stability, and in turn, better predict how this cast carbon pool responds to climate change, intensive agriculture, and land-use change [29]. Moreover, it is interesting to study results from FFC-NMR on dissolved organic matter (DOM). Understanding DOM structure and function is fundamental for comprehending global carbon cycling, contaminant transport and ocean chemistry in general [30].

The second area (interactions) investigates where and how contaminants bind in soils, sediments, and living species [31]. Understanding the interactions with soil and sediment will improve our knowledge of bioavailability and how to best proceed with remediation. Moreover, this learning can help to explain uptake, distribution, excretion, and bioaccumulation predicting more subtle long-term effects of exposure [21].

Another interesting application of the FFC-NMR technique is represented by the analysis of the porous structure of rocks that hold underground fluids. It could be possible to accurately estimate many petrophysical parameters, such as porosity, saturation, or permeability. Nowadays, borehole ^1H NMR is extensively used in oil and gas reservoir characterization, and recent developments have led to tools suitable for environmental applications [32].

Finally, a different field of application could be represented by food research. Nowadays, high-field NMR spectroscopy techniques are generally used to qualify and quantify the metabolites present in plant and animal tissues, to determine the composition and formulation of packaging materials, to food authentication, to optimize food processing parameters, to ensure microbial safety, and to inspect the physical and chemical quality of food [33, 34]. Despite the low resolution of low-field devices, the information obtained from FFC relaxation analysis is sufficient to study water content, compartments, diffusion, and movement [35]. The challenge is to find a well-stable model and numerical methods to robustly analyze the acquired NMRD profiles.

All these improvements and employments lead to increasing and enhancing efficiency in terms of energy and environment, but also in economic terms, because low-field devices have less environmental impact compared to high-field NMR devices considering energy consumption and the quantity of cryogenic liquids needed to keep magnets at low temperatures, aligning with sustainability goals [36].

Contributions and Outline

This thesis is structured into two parts, covering a total of 7 chapters. In Part I the background concepts are comprehensively presented and discussed. In Part II, the proposed methods to address the problem of interest are introduced and analyzed.

Part I is divided into 4 chapters:

- Chapter 1 introduces the principles of Nuclear Magnetic Resonance (NMR), focusing on nuclear spin dynamics and their interaction with magnetic fields. The chapter details the Fast Field Cycling (FFC) NMR technique, its historical evolution, and its use in analyzing Nuclear Magnetic Relaxation Dis-

persion (NMRD) profiles to study molecular dynamics. It concludes with a discussion of physical models for interpreting FFC data, emphasizing the model-free approach, which supports the numerical methods developed in this thesis.

- In Chapter 2, the essential regularization strategies and the optimization methods for developing the numerical methods proposed in later sections are described in a general context. In the first part, Tikhonov regularization for linear and non-linear problems is detailed [37]. The chapter concludes with the block non-linear Gauss-Seidel method, as proposed by Grippo and Sciandrone [38, 39], forming the foundation for the algorithm detailed in Chapter 6.

- In Chapter 3, the automatic techniques to compute the regularization parameter are described. Specifically, the first part of this chapter discusses the augmented Tikhonov framework, and it introduces the *Balancing Principle* (BP) [37], which is employed in the proposed methods later detailed. The chapter concludes with an introduction to the Uniform Penalty principle [40], based on L_2 regularization with locally adapted regularization parameters, providing the basis for strategies developed for NMRD profile analysis, and described later in Chapter 5.

- In Chapter 4, the mathematical foundations of machine learning are introduced, emphasizing its role in solving complex problems. The chapter then explores integrated models that merge traditional optimization with data-driven techniques. The final section focuses on the Plug-and-Play (PnP) method, highlighting its use of pre-trained denoisers within iterative algorithms. This framework's philosophy sets the stage for the novel method developed in this PhD work and described in Chapter 7.

Part II is divided into 3 chapters, presenting the proposed solution methods based on classical regularization and optimization strategies, and a hybrid technique based on the combination of a machine learning technique with a coordinate descent optimization algorithm:

- In Chapter 5, three approaches for analyzing NMRD profiles are proposed: *MF-UPen*, which uses locally adapted L_2 regularization; *MF-L1*, based on an L_1 penalty; and *MF-MUPen*, combining locally adapted L_2 and global L_1 penalties. All methods employ automatic parameter selection via the Balancing Principle and Uniform Penalty principles. Contributions include the implementation and testing of the MF-UPen and MF-MUPen algorithms, along with a dispersion analysis procedure to assess parameter estimation ranges. The chapter concludes with a comparison of algorithmic results, focusing on fit quality and correlation time distribution, supported by tests on representative NMRD profile datasets.

- In Chapter 6, a computational framework for automating FFC-NMR analysis is introduced, addressing the parameter identification problem for offset terms, correlation time distribution, and QRE parameters. This problem is formulated as a regularized non-linear least squares problem with box constraints, using an L_1-based regularization to promote sparse solutions. The estimation process employs an iterative approach where the regularization parameter is computed via the Balancing Principle, and the constrained optimization problem is solved using a two-block non-linear Gauss-Seidel method. The proposed method effectively separates the contributions of relaxation distributions and quadrupolar relaxation parameters, achieving accurate fits for NMRD profiles, including QRE-related local maxima. The chapter concludes with numerical experiments on synthetic and real data, demonstrating the robustness and efficiency of the algorithm.

- In Chapter 7, a machine learning framework is presented for NMRD profile inversion, focusing on the QRE phenomenon. A pre-trained feed-forward neural network, designed to predict QRE parameters, is

integrated into the NMRD data fitting process using the Plug-and-Play (PnP) approach. This method incorporates a novel loss function and a tailored training procedure, marking the first application of PnP techniques to FFC-NMR data analysis. Results are compared with robust optimization methods, showing strong agreement and highlighting the potential of neural networks to streamline the analysis of large datasets. The chapter concludes with numerical tests on food-related NMRD profiles, demonstrating the framework's effectiveness.

Related Publications

This thesis mainly refers to the following published papers:

- V. Bortolotti, P. Conte, G. Landi, P. Lo Meo, A. Nagmutdinova, G.V. Spinelli, and F. Zama. Robust Algorithms for the Analysis of Fast-Field-Cycling Nuclear Magnetic Resonance Dispersion Curves. *Computers*, 13(6), 129, 2024. https://doi.org/10.3390/computers13060129.
- G. Landi, G.V. Spinelli, F. Zama, D.C. Martino, P. Conte, P.L. Meo, and V. Bortolotti. An automatic L1-based regularization method for the analysis of FFC dispersion profiles with quadrupolar peaks. *Applied Mathematics and Computation*, 444:127809, 2023. https://doi.org/10.1016/j.amc.2022.127809.
- G.V. Spinelli, D. Evangelista, L. Hu, and F. Zama. Neural Network-Based Inversion of NMR Dispersion Profiles for Enhanced Analysis of Food Systems. *Neural Computing and Applications*, 2024. https://doi.org/10.1007/s00521-024-10859-y

Acknowledgment

Borsa di dottorato del Programma Operativo Nazionale Ricerca e Innovazione 2014-2020 (CCI 2014IT16M2OP005), risorse FSE REACT-EU, Azione IV.4 "Dottorati e contratti di ricerca su tematiche dell'innovazione" e Azione IV.5 "Dottorati su tematiche Green. CUP code: J35F21003200006.

Part I

Preliminaries

Chapter 1

Physics Background

This chapter lays the theoretical groundwork for understanding Nuclear Magnetic Resonance (NMR) and its advanced applications, comprehensively exploring the interplay between nuclear spin dynamics and magnetic fields. It begins with a discussion of the NMR phenomenon, delving into the quantum mechanical principles governing nuclear spin transitions and their interactions with external magnetic fields, which form the basis for signal detection and analysis. The second section introduces the Fast Field Cycling (FFC) NMR technique, an essential tool for this study. It traces its historical evolution and its role in mapping relaxation phenomena by analyzing Nuclear Magnetic Relaxation Dispersion (NMRD) profiles. These profiles serve as fingerprints of molecular dynamics, offering insights into interactions within bulk liquids and confined environments. The final section describes physical models used to interpret FFC data, with special attention given to the model-free approach, which is the basis for developing advanced numerical methods and strategies presented in this thesis work. This model avoids the constraints of predetermined assumptions about molecular dynamics, enabling the extraction of robust and adaptable insights across diverse systems. The chapter thus establishes a theoretical foundation for the innovative methodologies employed in subsequent analyses.

1.1 Nuclear Magnetic Resonance Phenomenon

Nuclear Magnetic Resonance (NMR) is a physical phenomenon in which atomic nuclei in a static magnetic field are perturbed by an externally applied Radio-Frequency (RF) pulse. This perturbation induces a resonant electromagnetic response from the nuclei, characterized by a frequency specific to the local magnetic environment experienced by the nucleus.

The process involves nuclear spin transitions between discrete quantum energy levels associated with angular momentum, with the energy differences typically on the order of thermal energy [1].

In 1938, Isidor Rabi extended the Stern-Gerlach experiment to measure nuclear magnetic resonance in molecular beams for the first time [41]. His pioneering work earned him the Nobel Prize in Physics.

Currently, NMR refers predominantly to phenomena in condensed matter systems, rather than the isolated atomic beams used in Rabi's experiments. Specifically, NMR today typically refers to nuclear spin transitions detected via energy absorption from RF pulses. The first experimental observations of NMR in condensed matter were independently made in 1946 by two research groups: E. M. Purcell, H. C. Torrey, and R. V. Pound at Harvard University, using paraffin [42], and F. Bloch, W. W. Hansen, and M. Packard at Stanford University, using water [43]. These discoveries led to Purcell and Bloch sharing the Nobel Prize in Physics in 1952.

[1] Energy in the range of thermal energy: $kT \sim 25\text{meV}$.

1.1.1 Resonance Condition

For the nuclear magnetic resonance phenomenon, atomic nuclei must possess a magnetic moment, enabling interaction with a typically constant and homogeneous magnetic field, known as polarizing or Zeeman field, and denoted as $\boldsymbol{B}_0$. This interaction leads to achieving an order between energy levels, influenced by the magnetic field and opposed by thermal agitation (kT).

When an atomic nucleus interacts with a uniform and constant magnetic field, $\boldsymbol{B}_0$, its nuclear magnetic moment, $\boldsymbol{\mu}$, tends to align with the direction of the magnetic field. Consequently, the external magnetic field exerts a torque on the magnetic moment:

$$\boldsymbol{\tau} = \boldsymbol{\mu} \times \boldsymbol{B}_0$$

This interaction causes the nuclear magnetic moment, $\boldsymbol{\mu}$, to precess around the direction of the magnetic field at a specific angular frequency, ν_0, known as the *Larmor frequency*. This frequency depends only on the nuclear species and on the intensity of the magnetic field $\boldsymbol{B}_0$. From this point onward, bold symbols will denote vector quantities, while non-bold symbols will represent their magnitudes.

On a macroscopic scale, this results in establishing an equilibrium nuclear magnetization, $\boldsymbol{M}$, which can be detected to extract information about the system.

The phenomenon consists of manipulating the magnetization by supplying energy to the system, which must satisfy the *resonance condition*. Using appropriate hardware, the evolution of the magnetization, as it returns to equilibrium, can be monitored.

In an NMR experiment, the signal is detected by a coil as an electric signal induced by the changing magnetic flux caused by the movement of the nuclear magnetization, $\boldsymbol{M}(t)$, in a rotating reference frame at the resonance frequency. The intensity of $\boldsymbol{M}(t)$ has the same units as the nuclear magnetic moment ([J/T]), which is proportional to the spin, $\boldsymbol{I}$, of the nucleus n with spin:

$$\boldsymbol{\mu} = \gamma_n \frac{h}{2\pi} \boldsymbol{I} \tag{1.1}$$

where γ_n is the gyromagnetic ratio of the nuclear species n, which characterized it (its unit is [MHz/T]) and h is the Planck's constant, i.e., $h = 6.63 \cdot 10^{-34}$ Js. Therefore, for nuclear magnetic resonance to occur, nuclei must have $\boldsymbol{I} \neq 0$.

In the following Table 1.1, the most common nuclear species employed in NMR experiments are reported with their characteristics, i.e., proton and neutron numbers, spin, and gyromagnetic ratio expressed as $\gamma/2\pi$.

Nuclei	Odd Protons	Odd Neutrons	Spin	$\gamma/2\pi$ [MHz/T]
^1H	1	0	1/2	42.58
^2H	1	1	1	6.54
^{14}N	1	1	1	3.08
^{13}C	0	1	1/2	10.71
^{19}F	1	0	1/2	40.08

Table 1.1: Characteristics of the most common nuclear species employed in NMR experiments.

For a system with a single spin, the quantum mechanical Hamiltonian operator representing the energy is given by $\boldsymbol{H}_z = -\gamma\hbar\boldsymbol{I}_z B_0$, where $\hbar$ is the reduced Planck's constant, i.e., $\hbar = h/2\pi = 1.05457 \cdot 10^{-34}$ Js.

In the presence of an external magnetic field, the *Zeeman effect* arises, which involves the splitting or shifting of spectral lines. The corresponding energy levels are determined by the eigenvalues of the Hamiltonian operator [44]:

$$E_m = -\gamma\hbar m B_0 \tag{1.2}$$

where $-I \leq m \leq +I$.

For instance, in the case of hydrogen atoms, the energy difference between the two energy levels, corresponding to transitions between them, is $\Delta E = \gamma \hbar B_0$. By supplying the system with electromagnetic waves of energy $h\nu_0$, matching ΔE, the *resonance condition* is satisfied, leading to the *Larmor frequency*:

$$\Delta E = \gamma \hbar B_0 = h\nu_0 = 2\pi\hbar\nu_0 = \hbar\omega_0 \qquad \Rightarrow \qquad \omega_0 = \gamma B_0$$

The ω_0 is the Larmor angular velocity, i.e., $\omega_0 = 2\pi\nu_0$. From now on, the Larmor resonance condition is identified by ω_0.

For example, for ^1H nuclei in a magnetic field $B = 1$ T, the frequency is $\nu = (\gamma/2\pi)B = 42.6$ MHz. Since this frequency falls in the megahertz range, it lies within the radio wave portion of the electromagnetic spectrum.

When a sample is placed in a region where a uniform magnetic field, $\boldsymbol{B}_0$, is applied, the splitting of energy levels can be observed.

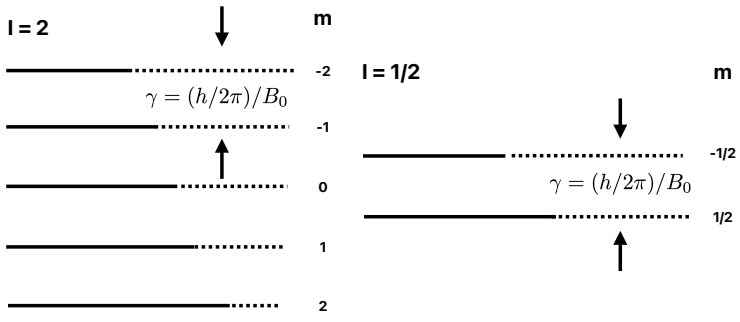


Figure 1.1: Energy level diagram illustrating spins under a Zeeman Hamiltonian. The left figure depicts a system with $I = 2$, while the right figure shows one with $I = 1/2$. The bold line represents the relative population of each state for an ensemble of systems in thermal equilibrium.

Nuclei are distributed across the $2I + 1$ energy levels according to the Boltzmann distribution, with the lower energy levels being the most populated

$$\frac{n_{m-1}}{n_m} = \exp\left\{-\frac{\hbar\omega_0}{kT}\right\}$$

where k is the Boltzmann's constant (i.e., $k = 1.38 \cdot 10^{-23}$ J/K). In a magnetic field $\boldsymbol{B}_0$ of magnitude 2 T, corresponding to a frequency of approximately 100 MHz, the exponent is on the order of 10^{-5}. This indicates that $kT \gg h\omega_0$, resulting in a population difference between energy levels that is proportional to the magnitude of B_0. The phenomenon is referred to as the *Curie law*.

Under this condition, the nuclear spin magnetization per volume unit is a vector aligned with the direction of $\boldsymbol{B}_0$; it has the same orientation, and its magnitude is given by:

$$M_0 = N\frac{\gamma^2\hbar^2 I(I+1)}{3kT}B_0 \tag{1.3}$$

This outcome can be derived by examining the energy levels' population. For simplicity, let us consider nuclei with a spin $I = 1/2$. In this case there are $2I + 1 = 2 \cdot \frac{1}{2} + 1 = 2$ possible states, with $m = +1/2$ (spin up) and $m = -1/2$ (spin down). The energy difference between these states is $\Delta E = \gamma\hbar B_0$ and the populations per volume unit of the two levels are denoted as N_+ and N_-. The total number of nuclei per volume unit is $N = N_+ + N_-$, which are distributed according to the Boltzmann distribution:

$$\frac{N_+}{N_-} = \exp\left\{\frac{\Delta E}{kT}\right\} = \exp\left\{\frac{\hbar\gamma B_0}{kT}\right\}$$

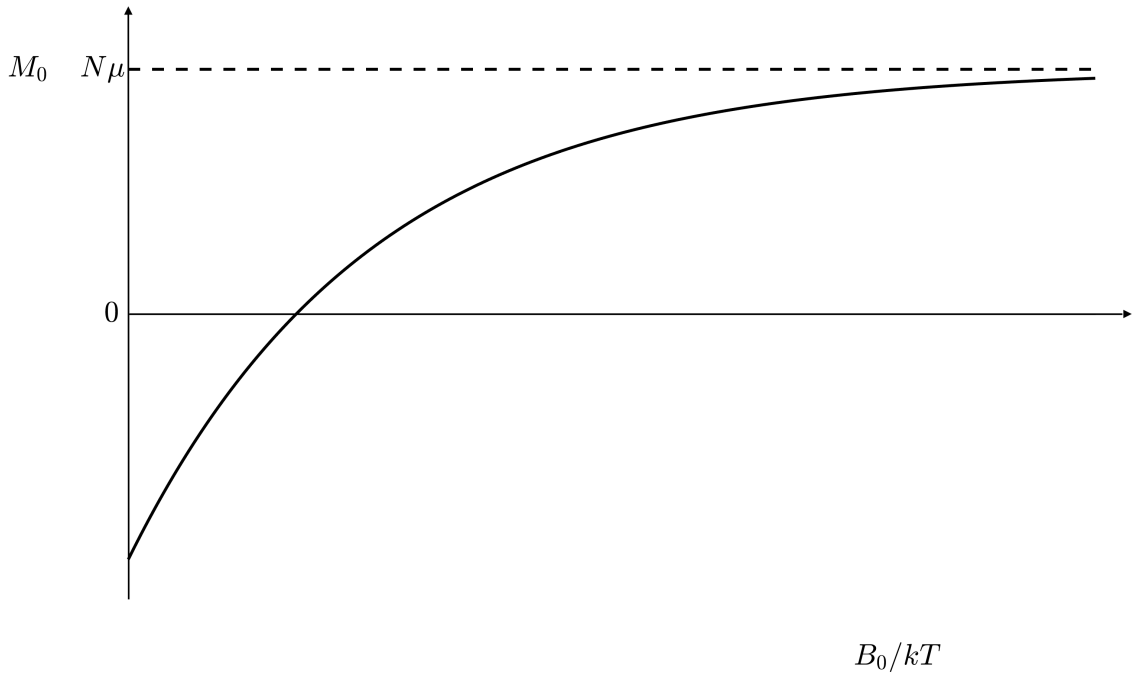


Figure 1.2: M_0 evolution. At a low-intensity field, M_0 is linearly proportional to B_0/kT, while at a higher-intensity field, M_0 saturates to $N\mu$, with all dipoles aligned with the field.

From the previous result, the populations per volume unit of the two levels follow:

$$N_+ = \frac{N}{1+\exp\left(-\gamma\hbar B_0/kT\right)} \qquad N_- = \frac{N}{1+\exp\left(+\gamma\hbar B_0/kT\right)}$$

and the equilibrium magnetization, M_0, is proportional to the surplus of spins up on spins down

$$M_0 = \mu(N_+ - N_-)$$

where μ is the nuclear magnetic moment. Hence

$$M_0 = N\mu\frac{1-\exp\left(-\gamma\hbar B_0/kT\right)}{1+\exp\left(-\gamma\hbar B_0/kT\right)} = N\mu\tanh\left(\frac{\gamma\hbar B_0}{2kT}\right)$$

Thus, M_0 is typically not directly proportional to the applied field. However, the dependence on B_0/kT remains linear at low magnetic fields, a condition that is generally upheld in experimental settings. At higher field strengths, the magnetization reaches saturation at $N\mu$, indicating that all magnetic dipoles are fully aligned with the field.

To meet this condition, i.e., $\hbar\gamma B_0 << 2kT$, the tanh function can be approximated using a first-order expansion:

$$M_0 = \frac{N\mu\hbar\gamma}{2kT}B_0 = \frac{N\hbar^2\gamma^2}{4kT}B_0$$

where $\mu = \hbar\gamma I = \frac{1}{2}\hbar\gamma$. If the spin is $I = 1/2$, the Curie law becomes (1.3).

Consequently, nuclei are distributed among the energy levels, primarily occupying the lower energy states. This results in an equilibrium magnetization per volume unit that aligns with the direction of $\boldsymbol{B}_0$. At the quantum level, when the system is irradiated with photons satisfying the resonance condition, the probabilities of absorption and stimulated emission are equal. However, since the lowest energy level is the

most populated, there is a net energy absorption, leading to a modification of the equilibrium magnetization vector. Information about the system can be obtained from the return of the nuclear magnetization to its equilibrium state following these changes.

1.1.2 Nuclear Spin Magnetization

Considering an NMR experiment, a spin ensemble behaviour can be described using the nuclear magnetization vector motion, which refers to the precession of $\boldsymbol{M}$ about a magnetic field according to the equation:

$$\frac{d\boldsymbol{M}}{dt} = \gamma \boldsymbol{M} \times \boldsymbol{B}_0$$

Precession happens when the vector $\boldsymbol{M}$ is not aligned with the polarizing magnetic field $\boldsymbol{B}_0$. To move $\boldsymbol{M}$ and excite the system, another RF magnetic field $\boldsymbol{B}_1$, which is perpendicular to $\boldsymbol{B}_0$ and oscillates at a frequency $\boldsymbol{\omega}_0$, needs to be applied. This allows for the measurement of how $\boldsymbol{M}$ returns to equilibrium, providing valuable information about the system.

Let us consider a $\boldsymbol{B}_0$ aligned to the z-axis of a reference frame, and let us apply a radio-frequency (RF) pulse $\boldsymbol{B}_1$ perpendicular to the polarizing field which precesses about z-axis with the same angular velocity of $\boldsymbol{M}$ (i.e., resonance condition). The magnetization $\boldsymbol{M}$ will move from the equilibrium of an angle proportional to the time and the amplitude of the RF pulse[2]. The angle is called *flip angle* and it is given by

$$\alpha = \gamma B_1 t$$

This process, described in the so-called *laboratory frame*, is shown in Figure 1.3, where the $\boldsymbol{M}$ precesses around $\boldsymbol{B}_0$ toward xy plane with a helical shape.

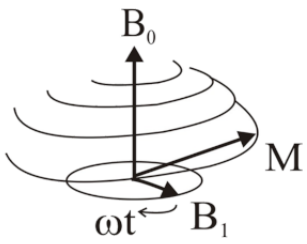
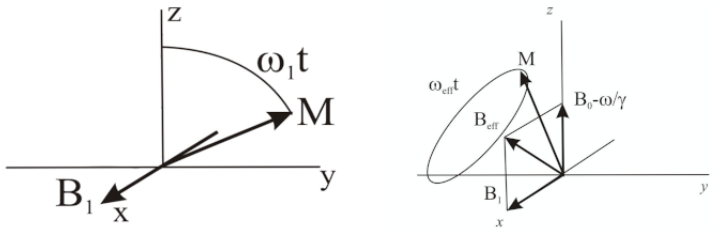


Figure 1.3: Evolution of $\boldsymbol{M}$ in the laboratory frame in the presence of a polarizing field $\boldsymbol{B}_0$, and a transverse RF pulse, $\boldsymbol{B}_1$. When $\boldsymbol{\omega} = \boldsymbol{\omega}_0$, $\boldsymbol{M}$ simultaneously precesses about $\boldsymbol{B}_0$ at $\boldsymbol{\omega}_0$, and about $\boldsymbol{B}_1$ at $\boldsymbol{\omega}_1$.

In this frame, the $\boldsymbol{M}$ components are

$$\begin{aligned} M_x(t) &= M_0 \sin(\omega_1 t) \sin(\omega_0 t) \\ M_y(t) &= M_0 \sin(\omega_1 t) \cos(\omega_0 t) \\ M_z(t) &= M_0 \cos(\omega_1 t) \end{aligned}$$

[2]The moving does not occur instantaneously, but lasts for the duration of the applied RF pulse.



(a) On-resonance RF pulse. (b) Off-resonance RF pulse.

Figure 1.4: Evolution of $\boldsymbol{M}$ in the rotating frame.

1.1.3 NMR signal and Relaxation

The vector $\boldsymbol{M}$ gradually precesses to 0 following an exponential process, hence the NMR signal detected by a coil is a damped sine wave, known as Free Induction Decay (FID). This signal is directly related to the intensity of the magnetization in the xy plane. The damping occurs because, once the spins are in the xy plane, they start to lose their phase coherence due to spin-spin interactions. The system tends to return to thermal equilibrium, characterized by

$$M_z(\text{equilibrium}) = M_0$$
$$M_{xy}(\text{equilibrium}) = 0$$

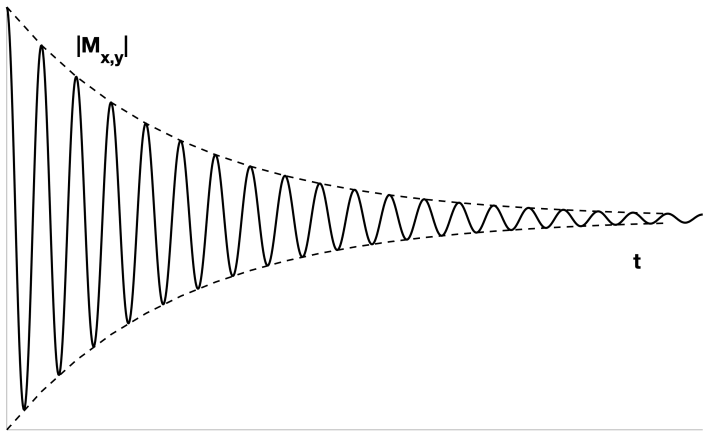


Figure 1.5: Free Induction Decay (FID).

This return to the equilibrium state is called *relaxation*, and the physical processes which regulate it after the application of an RF pulse are different for the longitudinal and the transverse magnetization components. For the longitudinal case, the process is an energetic process type, where the spins returning to the equilibrium yield their energy to the lattice, while, for the transverse case, the process is an entropic process type, which corresponds to decoherence of the transverse nuclear spin magnetization. The initial phase coherence of spins is lost until the phases are eventually disordered and there is no net M_{xy}. This is due to random fluctuations of the local magnetic field, which lead to random variations in the instantaneous NMR precession frequency of different spins.

The nuclear magnetization motion equation for an ensemble of free spins in the presence of a homogeneous field follows the so-called *Bloch equations*:

$$\frac{d\boldsymbol{M}}{dt} = \gamma \boldsymbol{M} \times \boldsymbol{B} \tag{1.4}$$

where $\boldsymbol{M}$ is the magnetization vector, $\boldsymbol{B}$ is the magnetic field vector, and γ is the gyromagnetic ratio of the specific nuclei under consideration.

Considering a polarizing static field applied along the z-axis, i.e., $B_z = B_0$, the return to equilibrium $M_z = M_0$ of the longitudinal magnetization, after shutting down the radio frequency pulse, depends on the velocity of the redistribution of spins on energy levels following Boltzmann distribution, so it is proportional to the difference between longitudinal equilibrium magnetization and longitudinal magnetization itself and it decreases in time:

$$\frac{dM_z(t)}{dt} = -\frac{M_z(t) - M_0}{T_1} \tag{1.5}$$

where T_1 is the longitudinal (or spin-lattice) relaxation time, which represents the decay constant for the recovery of the z component of the nuclear spin magnetization towards its thermal equilibrium value.

Similarly, for the transverse component, the return to equilibrium depends on the difference between its current value and its equilibrium value, which corresponds to 0. Hence, in the rotating frame,

$$\frac{dM_{xy}(t)}{dt} = -\frac{1}{T_2} M_{xy}(t) \tag{1.6}$$

where T_2 is the transverse (or spin-spin) relaxation time, and it refers to the decay constant for the component of $\boldsymbol{M}$ perpendicular to $\boldsymbol{B}_0$. Because T_2 relaxation only involves the phase coherence loss between spins, while T_1 relaxation involves an energy transfer and also a phase coherence loss, it always holds $T_2 \leq T_1$.

The evolution of the magnetization vector is described by the solutions of the Bloch equations (Figure 1.6).

$$M_z(t) = M_z(0)e^{-\frac{t}{T_1}} + M_0\left(1 - e^{-\frac{t}{T_1}}\right)$$
$$M_{xy}(t) = M_{xy}(0)e^{-\frac{t}{T_2}}$$

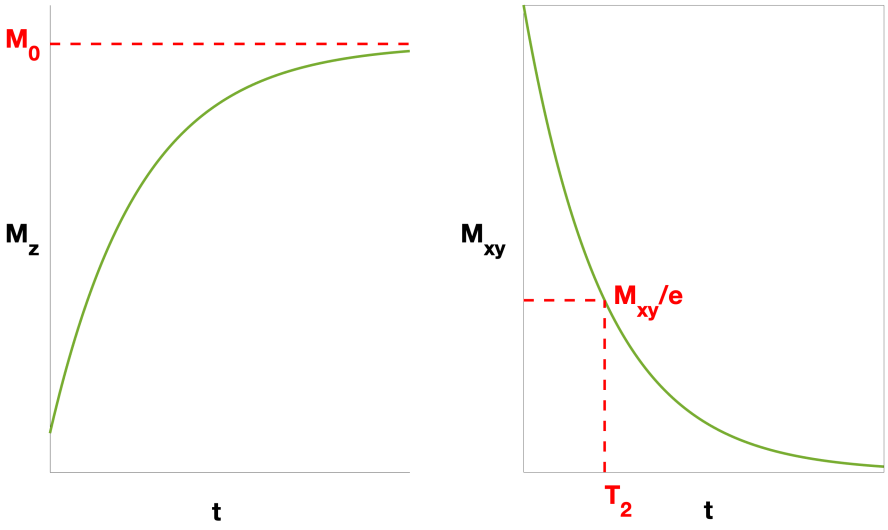


Figure 1.6: Evolution of the magnetization vector considering an RF pulse applied to have a flip angle $\alpha = -180°$. The initial value of the magnetization, $M(0) = -M_0$, hence $M_z(t) = M_0(1 - 2e^{-t/T_1})$.

In practice, considering the transverse relaxation, it is impossible to have a completely homogeneous polarizing field $\boldsymbol{B}_0$ applied in the entire volume of interest, therefore the inhomogeneities of the field contribute to dephasing. This depends on slight differences in the local magnetic field which lead spins to not precess with the same frequency, and it is manifested as a loss of signal. For this reason, the signal will relax faster to 0 with a decay constant smaller than T_2. This decoherence due to the inhomogeneities is not a true relaxation process, and it depends on the location of the molecules in the magnet. Considering molecules that are not moving (i.e., neglecting the diffusion process), the deviation from ideal relaxation is consistent over time, and the signal can be recovered by performing a specific NMR experiment.

The time constant in this case is known as T_2^*, and it usually is smaller than T_2

$$\frac{1}{T_2^*} = \frac{1}{T_2} + \frac{1}{T_{\text{inhom}}} = \frac{1}{T_2} + \gamma \Delta B_0$$

where ΔB_0 is the variation of the magnitude of the local field.

A FID is generated by applying one single pulse, but most of the NMR experiments, or sequences (i.e., a sequence of applied RF pulses), are based on the combination of different kinds of pulses.

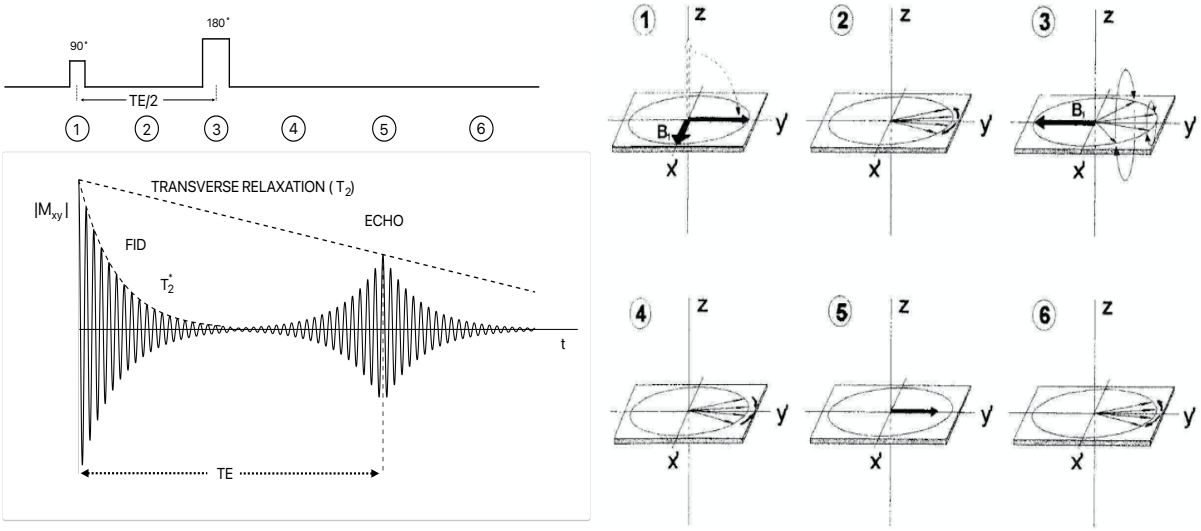
Applying two successive RF pulses a *Spin Echo* (SE) signal will be produced, and the time passing between the application of the first pulse and the peak of the spin echo signal is known as *Echo Time* (TE) [45].

The application of the second RF pulse causes a refocusing of spin phase information lost during the decay FID produced by the first RF pulse. This is possible because many of the T_2^* processes that produce the decay of the FID are symmetrically reversible, i.e., most of the signal after the first pulse is not destroyed, but becomes *disorganized*.

The generation of an echo is represented in Figure 1.7. Let us consider the vector representation (Figure 1.7 (b)). The first pulse is a 90° pulse (at Larmor frequency), which flips the spins into the xy plane (image (1)). The spins will start to lose coherence due to the inhomogeneities in the polarizing field, and some spin groups may precess faster than others (i.e., gaining phase) (image (2)). Successively, a 180° pulse is applied on the xy plane, rotating on the same plane and changing the verse of rotation of them (image (3)). Then, the spins will start to refocus (image (4)) reaching the peak of the spin echo signal (image (5)) at[3] t = TE and eventually, without the application of another pulse, spins will continue to rotate losing coherence and free decaying (image (6)). Spin echoes will form each time two successive pulses of any flip angle are applied to the system. The combination of pulses described before (90° and 180°) produces the maximum possible echo signal and it is the most employed one. The echo formed in this case is called *Hahn echo*.

The application of more than two pulses generates a *stimulated echo* (STE). Let us consider the Figure 1.8, where three pulses (indicated on the image with numbers, i.e., 1, 2, and 3) are applied to the system. The echoes A, B, and C are Hahn echoes, where A is generated by pulses 1, and 2; B is generated by pulses 2, and 3; and C is generated by pulses 1, and 3. The echo D is a secondary spin echo obtained by the application of the pulse 3 on spins of echo A. Specifically, in correspondence with the peak of echo A, spins are in phase in the transverse plane. Then, spins dephase with T_2^*. The pulse 3 (located exactly midway between A and D) affects some of these spins, allowing them to refocus into echo D. Finally, echo E is called stimulated echo. The generation in a vector spin diagram is shown in Figure 1.9, where only 4 spins (a, b, c, and d) are considered for simplicity: First, a 90° pulse is applied, flipping spins into xy plane (a); then the spins start to dephase (b). Spins are considered in pairs ($a-b$, $c-d$), where each one presents local fields offset of the same magnitude, but opposite polarities. Thus, a spin gains or loses phase corresponding to its partner at the same relative rate. The spins a and c move in local fields with lower intensity than $\boldsymbol{B}_0$, thus they precess slightly slower and lose phase, while b and d precess faster and gain phase because their

[3] TE corresponds to the double of the time between two RF pulses.



(a) (b)

Figure 1.7: Spin Echo generation.

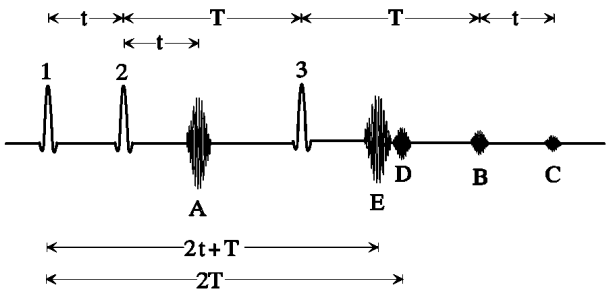


Figure 1.8: Sequence with 3 RF pulses applied and generation of a STE.

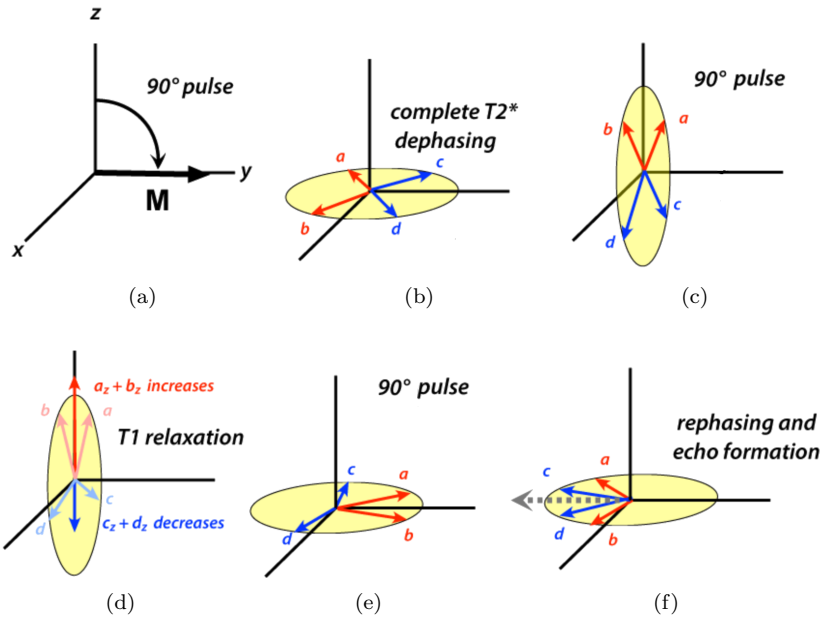


Figure 1.9: Generation of a stimulated echo from three 90° pulses.

local fields have higher intensity. Then, a second 90° pulse is applied (c) flipping the spins into xz plane. The z components remain aligned to the $\boldsymbol{B}_0$ direction, not precessing. The longitudinal relaxation occurs, having a net growth along $z-$axis (d), and for this reason, stimulated echo presents a T_1-weighting. The z components are flipped back to the xy plane due to the last 90° pulse, and they start to precess (e) rephasing along the y-axis and generating the stimulated echo (f). The amplitude of this echo is lower and it is spread out more widely in time because not all components rephase at the exactly same time.

Finally, increasing the number of pulses, n, leads to an exponential increase in the number of echoes, following $N_{\text{echoes}} = \frac{3^{n-1}-1}{2}$.

1.1.4 Steady-State solution of Bloch Equations

In an arbitrary homogeneous field, the equation of the motion of the nuclear magnetization for an ensemble of free spins is represented by (1.4). In a polarizing static field with $B_z = B_0$ applied along the $z-$axis, the trend of the magnetization towards its equilibrium value represented by $M_z = M_0 = \chi_0 B_0$ can often be described with good accuracy by (1.5). The variable χ_0 represents the magnetic susceptibility, and it can be derived from (1.3), i.e., $\chi_0 = \frac{N\gamma^2\hbar^2 I(I+1)}{3kT}$. Moreover, if an RF pulse is applied such that the magnetization gains a component perpendicular to the magnetic field $\boldsymbol{B}_0$, the local magnetic field variations, arising from spin-spin and spin-lattice interactions, lead to a decay of the transverse magnetization at a rate characterized by (1.6). Alternatively, this decay can be described by considering its two components separately:

$$\frac{dM_x}{dt} = -\frac{M_x}{T_2}$$
$$\frac{dM_y}{dt} = -\frac{M_y}{T_2}$$

Finally, in the presence of an applied field, i.e., the sum of the polarizing field and a much smaller radio-frequency field, the motion due to the relaxation can be superposed on the motion of the free spins, leading to the following equation:

$$\frac{d\boldsymbol{M}}{dt} = \gamma \boldsymbol{M} \times \boldsymbol{B} - \frac{M_x \boldsymbol{i}' + M_y \boldsymbol{j}'}{T_2} - \frac{M_z - M_0}{T_1}\boldsymbol{k}' \tag{1.7}$$

where $\boldsymbol{i}'$, $\boldsymbol{j}'$, and $\boldsymbol{k}'$ are the unit vectors of the laboratory frame of reference.

Let us assume that the applied field is the sum of a polarizing field $\boldsymbol{B}_0$ applied along the z-axis, $B_z = B_0 = -\frac{\omega}{\gamma}$, and of an RF field $\boldsymbol{B}_1$ of amplitude $B_1 = -\frac{\omega_1}{\gamma}$ rotating at a frequency ω in the neighbourhood of ω_0. The RF field will be usually one of the components of the rotating applied field $B_x = 2B_1 \cos(\omega t)$ linearly polarized along the x-axis of the laboratory frame, neglecting the effect of the counter-rotating component.

The effective field in the rotating frame is

$$\boldsymbol{B}_{\text{eff}} = \left(B_0 + \frac{\omega}{\gamma}\right)\boldsymbol{k} + B_1\boldsymbol{i} = \frac{(\omega - \omega_0)\,\boldsymbol{k} - \omega_1\boldsymbol{i}}{\gamma} = \frac{\Delta\omega\boldsymbol{k} - \omega_1\boldsymbol{i}}{\gamma}$$

where $\boldsymbol{i}$, $\boldsymbol{j}$, and $\boldsymbol{k} = \boldsymbol{k}'$ are the unit vectors of the rotating frame, and $\omega_0 = -\gamma B_0$, $\omega_1 = -\gamma B_1$. The equation (1.7) in the rotating frame becomes

$$\frac{d\boldsymbol{M}}{dt} = \gamma\left(\boldsymbol{M} \times \boldsymbol{B}_{\text{eff}}\right) - \frac{\tilde{M}_x\boldsymbol{i} + \tilde{M}_y\boldsymbol{j}}{T_2} - \frac{M_z - M_0}{T_1}\boldsymbol{k} \tag{1.8}$$

where $\tilde{M}_x$, and $\tilde{M}_y$ are the components of $\boldsymbol{M}$ in the transverse plane in that frame. Equation (1.8) can be rewritten as

$$\begin{array}{l} \frac{d\tilde{M}_x}{dt} = -\frac{\tilde{M}_x}{T_2} + \Delta\omega\tilde{M}_y \\ \frac{d\tilde{M}_y}{dt} = -\Delta\omega\tilde{M}_x - \frac{\tilde{M}_y}{T_2} - \omega_1 M_z \\ \frac{d\tilde{M}_z}{dt} = \omega_1\tilde{M}_y - \frac{M_z - M_0}{T_1} \end{array} \tag{1.9}$$

where $\Delta\omega = (\omega - \omega_0)$.

For fixed values of the parameters, the solution of (1.9) is a sum of decreasing exponential terms and of a steady-state solution obtained by setting the following condition

$$\frac{d\tilde{M}_x}{dt} = \frac{d\tilde{M}_y}{dt} = \frac{dM_z}{dt} = 0$$

and, after a sufficiently long time for the transient exponentials to have decayed, the steady solution can be written as [46]

$$\begin{array}{l} \tilde{M}_x = -\frac{\Delta\omega\gamma B_1 T_2^2}{1+(T_2\Delta\omega)^2+\gamma^2 B_1^2 T_1 T_2} M_0 \\ \tilde{M}_y = -\frac{\gamma B_1 T_2}{1+(T_2\Delta\omega)^2+\gamma^2 B_1^2 T_1 T_2} M_0 \\ M_z = -\frac{1+(\Delta\omega T_2)^2}{1+(T_2\Delta\omega)^2+\gamma^2 B_1^2 T_1 T_2} M_0 \end{array} \tag{1.10}$$

The transverse components in the laboratory frame can be derived starting from the rotating frame ones following

$$M_x + iM_y = \left(\tilde{M}_x + i\tilde{M}_y\right)e^{i\omega t} \Longrightarrow \begin{cases} M_x = \tilde{M}_x\cos(\omega t) - \tilde{M}_y\sin(\omega t) \\ M_y = \tilde{M}_x\sin(\omega t) + \tilde{M}_y\cos(\omega t) \end{cases}$$

The three components of $\boldsymbol{M}$ are proportional to M_0 because, in the absence of initial polarization, i.e., of populations' inequalities between magnetic energy levels, the nuclear magnetic phenomenon cannot be observed.

The solutions (1.10) can be geometrically represented, where the locus of the vector $\boldsymbol{M}$ as a function of B_1, and $\Delta\omega$ is an ellipsoid with axes $2a_x = 2a_y = M_0\sqrt{T_2/T_1}$, $2a_z = M_0$, and centred at the point $x = y = 0$, $z = \frac{1}{2}M_0$.

1.1.5 Modified Bloch equations in low fields

It is interesting to discuss how nuclear spin magnetization (and thus the signal) behaves in the presence of low-intensity fields, considering that the Fast Field Cycling NMR sequence is a low-field technique.

When the intensity of the polarizing static field is comparable either to the radio-frequency one or to $1/\gamma T_2$, the Bloch equations (1.7) and their steady-state solutions (1.10) must be modified. Specifically, if a radio-frequency field $B_1(t)$ is applied, the assumption of magnetization relaxing towards the equilibrium value $M_0 = \chi_0 B_0$ should be replaced by that of relaxation towards the instantaneous value $\chi_0(B_0 + B_1(t))$. If B_1 is not small compared with B_0, the condition of relaxation towards the value $\chi_0(B_0 + B_1(t))$ is such that the following assumption holds $T_1 = T_2 = T$ [46].

Therefore, the modified Bloch equations can be rewritten as follows:

$$\frac{d\boldsymbol{M}}{dt} = \gamma \boldsymbol{M} \times \boldsymbol{B} - \frac{\boldsymbol{M} - \chi_0 \boldsymbol{B}}{T} \tag{1.11}$$

Now, it is necessary to distinguish between the response to an oscillating field and to a rotating field, particularly considering the counter-rotating component in cases where $\boldsymbol{B}_1$ is comparable in magnitude to $\boldsymbol{B}_0$.

The equations (1.9) in the rotating field can be rewritten as follows:

$$\begin{array}{l} \frac{d\tilde{M}_x}{dt} = -\frac{\tilde{M}_x - \chi_0 B_1}{T} + \Delta\omega \tilde{M}_y \\ \frac{d\tilde{M}_y}{dt} = -\Delta\omega \tilde{M}_x - \frac{\tilde{M}_y}{T} - \omega_1 M_z \\ \frac{dM_z}{dt} = \omega_1 \tilde{M}_y - \frac{M_z - M_0}{T} \end{array} \tag{1.12}$$

Following the formalism described in [46], the steady-state solutions of (1.12) can be expressed using the RF susceptibilities $\chi_R^{'M}(\omega)$, and $\chi_R^{''M}(\omega)$, where R stands for rotating and M identifies the modified Bloch equations. The susceptibilities $\chi'(\omega)$, and $\chi''(\omega)$ represent the real and the imaginary parts of the RF susceptibility $\chi = \chi' - i\chi''$, defined by the relations $B_x = 2B_1 \mathrm{Re}\{e^{i\omega t}\}$, $M_x = 2B_1 \mathrm{Re}\{e^{i\omega t}\}$

$$\begin{array}{l} \chi_R^{'M} = \frac{\tilde{M}_x}{2B_1} = -\frac{1}{2}\frac{\omega_0 \Delta\omega T^2 - \left[1 + (\gamma B_1 T)^2\right]}{1 + (T\Delta\omega)^2 + (\gamma B_1 T)^2}\chi_0 \\ \chi_R^{''M} = -\frac{\tilde{M}_y}{2B_1} = \frac{1}{2}\frac{\omega T}{1 + (T\Delta\omega)^2 + (\gamma B_1 T)^2}\chi_0 \end{array} \tag{1.13}$$

In case of negligible saturation ($|\gamma| B_1 T << 1$), the equation (1.13) becomes

$$\chi_R^{'M} = \chi_R^{'} + \frac{1}{\omega_0 T}\chi_R^{''}, \qquad \chi_R^{''M} = \chi_R^{''} - \frac{1}{\omega_0 T}\chi_R^{'}$$

where

$$\chi_R^{'} = -\frac{\chi_0}{2}\frac{\omega_0 \Delta\omega T^2}{1 + (T\Delta\omega)^2}, \qquad \chi_R^{''} = \frac{\chi_0}{2}\frac{\omega_0 T}{1 + (T\Delta\omega)^2}$$

are the rotating field susceptibilities for the unmodified Bloch equations ((1.7)). Following this, M_z will be

$$M_z = \chi_0 \left[B_0 + \frac{(\omega/\gamma)(\gamma B_1 T)^2}{1 + (T\Delta\omega)^2 + (\gamma B_1 T)^2} \right] \tag{1.14}$$

This result is in contrast to the result obtained in the unmodified case (1.10), which can be rewritten as

$$M_z = \chi_0 \left[B_0 + \frac{(\omega_0/\gamma)(\gamma B_1 T)^2}{1 + (T\Delta\omega)^2 + (\gamma B_1 T)^2} \right] \tag{1.15}$$

A comparison of these last results for M_z reveals that when $B_0 = -\omega_0/\gamma = 0$, equation (1.14) predicts a non-zero steady-state magnetization, in contrast to equation (1.15). This outcome makes sense from a physics perspective: in the rotating frame, the spins experience an effective static field B_{eff} with components $B_z = -\omega/\gamma$, $B_x = B_1$, around which they precess, while gradually relaxing towards $M_x^0 = \chi_0 B_1$. This leads to an asymmetry between positive and negative values of M_z, creating a non-zero steady-state value for it. The theoretical predictions from equation (1.14) have been experimentally verified [47].

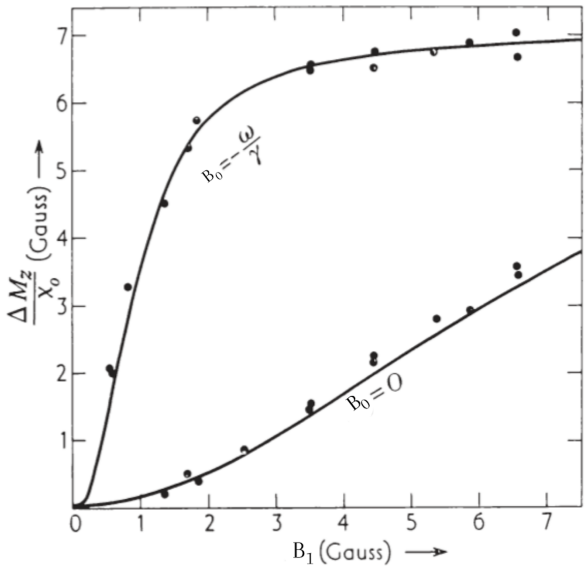


Figure 1.10: Observed signal in a static field B_0 as a function of RF amplitude in solid diphenyl, picril hydrazil. The full curves give the theoretical prediction of equation (1.14), assuming $T_1 = T_2 = T = 6.2{\cdot}10^{-8}\ s$, measured independently [46].

The Figure 1.10 [46] presents $\Delta M_z = M_z - \chi_0 B_0$ predicted by equation (1.14) as a function of the rotating field's strength B_1 for the cases $B_0 = -\omega/\gamma$, and $B_0 = 0$, with experimental points. The agreement between theory and experiment confirms that the unmodified Bloch equations which predict $M_z = 0$ when $B_0 = 0$ are inaccurate.

Let us consider a simple specific case where $B_0 = 0$. The modified Bloch equation (1.11) can be rewritten as

$$\begin{aligned} \tfrac{dM_x}{dt} &= -\tfrac{M_x - \chi_0 2B_1 \cos(\omega t)}{T} \\ \tfrac{dM_y}{dt} &= 2\gamma M_z B_1 \cos(\omega t) - \tfrac{M_y}{T} \\ \tfrac{dM_z}{dt} &= -2\gamma M_y B_1 \cos(\omega t) - \tfrac{M_z}{T} \end{aligned} \tag{1.16}$$

The exact steady-state solution of the previous system (1.16) is immediately obtained

$$\begin{aligned} M_y &= M_z = 0 \\ M_x &= 2\chi_0 B_1 \tfrac{\cos(\omega t) + (\omega T)\sin(\omega t)}{1 + (\omega T)^2} \end{aligned}$$

In a more general low-field case, with $B_z \neq 0$ and an RF field $B_x = 2B_1 \cos(\omega t)$ the system (1.11) cannot be analytically solved, hence numerical approaches have been used to study the evolution of the magnetization in low-field conditions.

1.2 Fast Field Cycling NMR Technique

This section provides an in-depth examination of the Fast Field Cycling (FFC) NMR Technique, which is essential for this thesis's data collection and analysis. The discussion begins with an explanation of FFC as the technique from which data is derived. It highlights its critical role in developing the numerical methods and strategies employed to interpret Nuclear Magnetic Relaxation Dispersion (NMRD) profiles. By elucidating the principles and historical context of FFC, this section establishes a framework for understanding how the acquired data informs the models utilized in subsequent analyses. The unique characteristics of NMRD profiles, which serve as distinct fingerprints for different materials, further underscore the relevance

of FFC in advancing research across various scientific disciplines, including material science, food science, and pharmaceuticals [16, 17, 18, 22].

1.2.1 Introduction and Historical Outline

The analysis of the longitudinal (spin-lattice) relaxation rate, i.e., the inverse of the longitudinal relaxation time, on the polarizing B_0 field provides exhaustive information into events at the molecular level beyond the reach of many other techniques [48, 1, 49, 50, 51, 52].

The Field Cycling (FC) technique allows measurements of the nuclear magnetic relaxation rates, as a function of the Larmor angular frequency ($\omega_0 = 2\pi\nu_0 = \gamma B_0$), thus mapping the spectral energy distribution of all the molecular motions affecting magnetic moments.

These methods were introduced shortly after the NMR phenomenon's discovery, but their development has been slower than other techniques, e.g., imaging and spectroscopy. The reasons lie in the complex nature of the NMR relaxation theories, and molecular dynamics models to analyze the acquired Nuclear Magnetic Relaxation Dispersion (NMRD) profiles (i.e., the relaxation rate profile), but also in technical challenges in building optimized equipment.

Significant molecular dynamics effects are observed at low magnetic fields, where the relaxation times tend to become shorter at lower fields, and the intensity of the signal generated by a system of spins in a B_0 field approximately decreases as $B_0^{3/2}$. In this regimen (i.e., low B_0, with Larmor frequency on the order of a few kHz), the local field may become comparable or higher than B_0, and the T_1 may be shorter than the longest correlation time, τ_c, hence the "weak collision" condition is violated (i.e., it is expected $T_1 \gg \tau_c$ when the fluctuating perturbations are of small intensity compared to the splitting energy due to B_0).

Moreover, in this low field condition, for nuclei with spins 1/2, the predominant longitudinal relaxation mechanism depends on the fluctuations due to dipole-dipole couplings, while for spins 1, nuclei with their electric quadrupole moment couple with molecular electric field local gradients, dominating the relaxation process.

The most powerful characteristic of the NMRD profiles is that they are unique for each material or substance, resulting in a single *fingerprint* for material identification. Fast Field Cycling applies to many fields, from research to industry, including food science applications [2, 7], material science [5, 6], and pharmaceutical [4] to name a few.

Historically, the first experiments regarding the FC approach were performed by N.F. Ramsey and R.V. Pound at Harvard, in 1949-1951. Then, between the 1950s and 1970s, this technique was improved by many groups of researchers mostly in the United States. The developed technique consisted of manually or mechanically moving the sample between two magnets (i.e., shuttling processes). The shuttling process is cyclically repeated, and for this reason, the technique was called field-cycling NMR relaxometry. Later, a new approach was proposed consisting of holding still the sample and switching the magnetic field using an air core electromagnet, speeding up the procedure. This led to calling the new method Fast Field-Cycling NMR relaxometry, which was first investigated by A.G. Redfield, F. Noack, S.H. Koenig, and R. Kimmich. Finally, the first commercial FFC-NMR relaxometer based on the Noack-Schweikert technology was produced in Italy by Stelar company in 1996. Since then, several improvements in both hardware and software have been introduced to extend the range of applications [53].

1.2.2 Quantum Formalism

From a quantum perspective, the nuclear spins' dynamics can be studied using the time-dependent Shrödinger equation, where all the information about the spin states, positions, and velocities of electrons and nuclei, is

completely described by a wave function $|\Psi\rangle$[4]. To maintain consistency with standard quantum mechanical notation, the wave function $|\Psi\rangle$ is represented using the bra-ket formalism, where $|\Psi\rangle$ denotes a vector in a Hilbert space. Despite the Shrödinger equation being complete, it cannot be used in practice, thus it is necessary to introduce a simplified approach, the *spin Hamiltonian hypothesis*, which only uses the nuclear spin states.

$$\frac{d}{dt}\left|\Psi_{\text{spin}}(t)\right\rangle \approx -\frac{i}{\hbar}\mathcal{H}^{\text{spin}}\left|\Psi_{\text{spin}}(t)\right\rangle$$

where $\mathcal{H}^{\text{spin}}$ represents the nuclear spin Hamiltonian, and $|\Psi_{\text{spin}}\rangle$ is the wave function of the spin state of the nuclei. Henceforth, for simplicity, $|\Psi\rangle$ will represent the nuclear spin state, and $\mathcal{H}$ is the nuclear spin Hamiltonian.

This approximation relies on the concept of timescale separation, which implies statistical independence between nuclear and electronic movements. Essentially, the motion of electrons is so fast that nuclear spins are influenced only by the average effect of the fields produced by the electrons over time. Moreover, the energies associated with nuclear spins are typically too small to have any impact on the behaviour of the electrons or the overall motion of the molecules. Thus, it is reasonable to assume that, under typical temperature conditions, the approximation of the nuclear spin Hamiltonian holds for nearly all systems.

The eigenvalues of the quantum mechanics operator $\mathcal{H}$ are the energy values of the quantum system, while the eigenfunctions form an orthonormal basis $|n\rangle$ (i.e., an eigenbasis)

$$\mathcal{H}\left|n\right\rangle = E_n\left|n\right\rangle$$

in which E_n represents the energy level of the steady state $|n\rangle$. Moreover, the wave function of a spin system can be represented as a linear combination of eigenbasis

$$\left|\Psi\right\rangle = \sum_n c_n\left|n\right\rangle$$

In this equation, the c_n coefficients are associated with the probability of the system existing in the state $|n\rangle$. This equation illustrates the principle of state superposition, which is a key concept in NMR theory.

Electric and Magnetic spin Hamiltonian

Using solely symmetry principles, without complex calculations, it can be demonstrated that electric interactions are non-existent for nuclei with spin $I = 1/2$, $\mathcal{H}^{\text{ele}} = 0$. Specifically, for these spin-1/2 nuclei, no electrical energy terms rely on the nucleus's internal structure or orientation. However, for nuclei with spin $I > 1/2$, the electric charge distribution lacks spherical symmetry, causing the nucleus's electric energy to vary based on its orientation relative to the molecule's structure. Consequently, the nuclear electric quadrupole moment interacts with the surrounding electric field gradient, resulting in an electric quadrupole interaction term in the spin Hamiltonian $\mathcal{H}^{\text{ele}} = \mathcal{H}^{\mathcal{Q}}$.

The quadrupole interaction is influenced by both the nuclear quadrupole moment and the electric field gradient, which is determined by the molecular environment. This makes the NMR of quadrupolar nuclei more complex than that of spins-1/2, as both electric and magnetic factors affect molecular dynamics.

Recalling (1.1) and (1.2), the quantum expression for the energy of a magnetic moment interacting with a magnetic field $\boldsymbol{B}$ is $\mathcal{H}^{\text{mag}} = -\boldsymbol{\mu}\cdot\boldsymbol{B}$.

The magnetic energy is lowest when $\boldsymbol{\mu}$ is parallel to $\boldsymbol{B}$ and highest when they are antiparallel. The magnetic and electric fields affecting a nuclear spin can originate from external equipment (external spin interactions) or the sample itself (internal spin interactions). External spin interactions are exclusively

[4] Alternatively, one may adopt the Heisenberg formalism, wherein the time dependence is embedded in the operators, instead of wave functions.

magnetic, with various applied magnetic fields used to manipulate nuclear spins. For spin-1/2, internal spin interactions are purely magnetic, while spins $I > 1/2$ also experience electric quadrupolar interactions. Nuclear spins typically couple more strongly with external apparatus than with the molecular environment, as external interactions are generally more intense.

In NMR relaxometry, the external perturbations are not weak, and this means that the measuring instrument becomes, in a sense, part of the system under investigation.

External Spin Interactions

As previously described in Section 1.1, in an NMR experiment a polarizing homogeneous and static magnetic field labelled as $\boldsymbol{B}_0$ is applied to the sample. This can be generated by a permanent, electromagnetic, or superconducting magnet. The system is excited using RF oscillating fields, labelled as $\boldsymbol{B}_1$. Generally, these pulses are generated by a coil in the probe and are as spatially homogeneous as possible and perpendicular to $\boldsymbol{B}_0$. Moreover, depending on the experiment, some spectrometers also present gradient coil, which generates a magnetic field gradient, labelled as $\boldsymbol{B}_{\text{grad}}(\boldsymbol{r}, t)$, and which depend on the position and can also depend on time. Usually $\boldsymbol{B}_{\text{grad}} \ll \boldsymbol{B}_0$.

Hence, considering the Hamiltonian formulation, the external part is

$$\mathcal{H}^{\text{ext}} = \mathcal{H}^{\text{static}} + \mathcal{H}^{\text{RF}}(t) + \mathcal{H}^{\text{grad}}(\boldsymbol{r}, t)$$

Considering $\mathcal{H}_i^{\text{static}}$ the interactions between $\boldsymbol{B}_0$ and the spins I_i, the previous total spin Hamiltonian related to the static field will be $\mathcal{H}^{\text{static}} = \sum_i \mathcal{H}_i^{\text{static}}$. Similarly, the other terms of the Hamiltonian in (1.2.2) describe the interactions between spins and $\boldsymbol{B}_1$, and $\boldsymbol{B}_{\text{grad}}$.

From 1.1.1, defining $\omega_i = -\gamma_i B_0$ the Larmor angular frequency of spin I_i, the Hamiltonian term corresponding to the static field (directed along the $z-$axis) is

$$\mathcal{H}_i^{\text{static}} = -\gamma_i B_0 I_{i_z}$$

while, the term related to the RF pulse, during the application, can be approximated with:

$$\mathcal{H}_i^{\text{RF}}(t) = -\omega_i^{\text{nut}} \left[\cos\left(\omega_{\text{ref}} t + \varphi\right) I_{i_x} + \sin\left(\omega_{\text{ref}} t + \varphi\right) I_{i_y}\right]$$

where ω_i^{nut} represents the nutation frequency, i.e., a measure of the RF intensity experienced by spins in angular frequency units, ω_{ref} represents the spectrometer reference frequency, and φ is the initial phase of the RF field.

Finally, considering an applied external gradient field directed along the three laboratory frame axes, the intensity is usually much lower than the polarizing one, thus, the transverse component can be neglected, and e.g., the correspondent Hamiltonian for a gradient along $x-$axis is

$$\mathcal{H}_i^{\text{grad}}(\boldsymbol{r}, t) = -\gamma_i G_x(t) x I_{i_z}$$

In the same way, for G_y, and G_z.

Internal Spin Interactions

Generally, the spin-spin couplings cannot be neglected in most NMR experiments and for most samples. The effect of the electromagnetic fields generated from the sample itself on the nuclei can be represented with the internal Hamiltonian, i.e.,

$$\mathcal{H}^{\text{int}} = f \sum_k F^{(k)} O^{(k)}$$

where f represents a specific constant depending on the interaction, $F^{(k)}$ are the functions related to the fluctuating structure, i.e., random functions of the relative positions of two spins, and $O^{(k)}$ represents the spin-operator functions acting on the spin variables. For instance, $F^{(k)}$ in the case of intramolecular Hamiltonian dipolar coupling (where the interdipole distance can be considered constant), can be described through the second-order spherical harmonics $Y_{2,m}(\theta, \varphi)$, with $m = 0, \pm1, \pm2$, and $\theta(t)$, and $\varphi(t)$ the polar and the azimuthal angles respectively.

Assuming the substances in question are diamagnetic -that is, devoid of unpaired electron spins (specifically, free of paramagnetic impurities)- the list of relevant spin interactions can be summarized as follows:

- *Direct dipole-dipole coupling* (Figure 1.11 (a)) refers to the immediate interactions between nuclear spins, which occur independently of the surrounding molecular electron cloud. Each nucleus possesses a magnetic dipole moment, generating a magnetic field that influences other nuclear magnetic moments. This interaction can be classified as intradipolar when it involves nuclei within the same molecule, or interdipolar when it occurs between nuclei of different molecules. Additionally, these interactions can be homonuclear if the participating nuclei are of the same type or heteronuclear if they differ in species.

- *Quadrupolar coupling* (Figure 1.11 (b)) refers to the interactions of an electric nature between nuclei that possess a spin greater than 1/2 and the electric fields in their vicinity. Unlike simpler nuclear spins, these nuclei have a charge distribution that extends over a volume, allowing them to interact with the gradient of the electric field generated by surrounding electron clouds. This type of coupling is significant in understanding the behaviour of certain nuclei in various environments, as it reveals how their electric quadrupole moments respond to external electric field variations. By examining quadrupolar coupling, valuable insights can be obtained into molecular symmetry, dynamics, and the local electronic environment surrounding these nuclei.

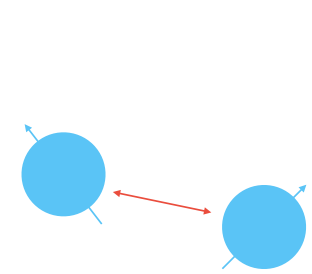
- *Chemical shift* (Figure 1.11 (c)) refers to the indirect interaction between the external magnetic field and nuclear spins, mediated by the electrons that orbit the nuclei. This interaction can lead to a subtle shift in the resonance frequency of the nuclei. Chemical shifts can be classified as isotropic or anisotropic, with the latter known as Chemical Shift Anisotropy (CSA). In an isotropic scenario, the orientation of the sample relative to the magnetic field does not influence the chemical shift; it is solely dependent on the strength of the applied magnetic field (B_0). Conversely, in anisotropic cases, the orientation of the sample can significantly affect the observed shifts, providing deeper insights into molecular structure and dynamics.

- *J-coupling* (Figure 1.11 (d)), also known as scalar coupling, refers to the indirect interactions between nuclear spins that occur through the mediation of valence electrons. Unlike other types of coupling, J-coupling is not influenced by the strength of the external magnetic field (B_0).

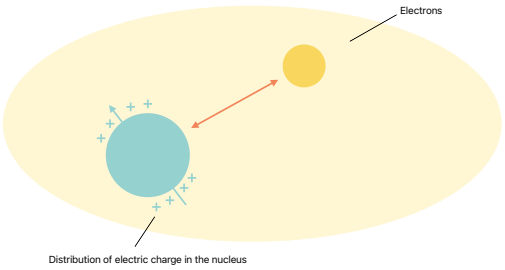
- *Spin–rotation interaction* (Figure 1.11 (e)) refers to the coupling between the nuclear magnetic moment and the magnetic fields produced by the electric currents that arise from the rotational motion of molecules. This interaction is significant in understanding how molecular rotation influences nuclear spins. As molecules rotate, they generate varying magnetic fields, which can affect the behaviour of nearby nuclear spins. This interplay provides valuable insights into molecular dynamics and structure.

Quadrupole coupling is generally the most significant interaction in terms of internal spin interactions; however, it vanishes for nuclei with a spin of 1/2.

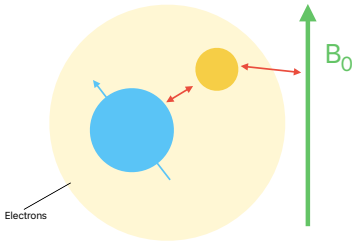
The mathematical description of nuclear spin interactions can be complex. Nevertheless, a more straightforward version of the internal Hamiltonian is frequently applicable, especially under the assumption of a very strong external magnetic field -this is referred to as the secular approximation- and when molecular motion is rapid, known as motional averaging. The parts of the Hamiltonian that are typically excluded pertain



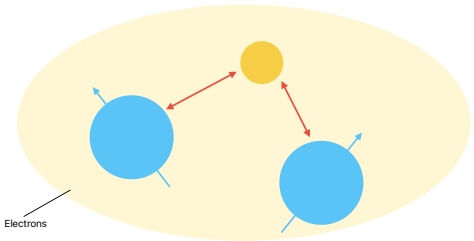
(a) Direct dipole-dipole coupling.



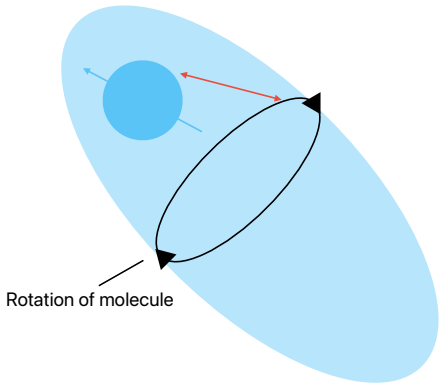
(b) Quadrupolar coupling.



(c) Chemical shift.



(d) J-coupling.



(e) Spin-rotation interaction.

Figure 1.11: Internal spin interactions. The yellow part represents the electron cloud.

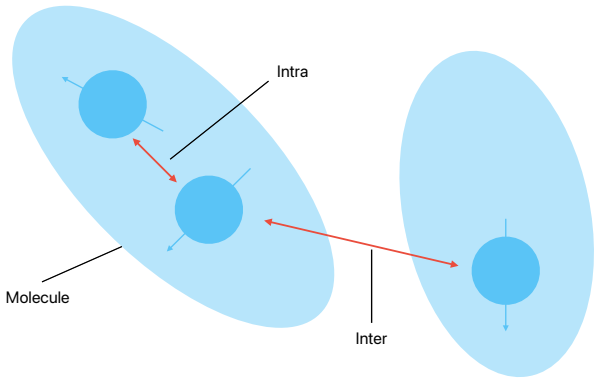


Figure 1.12: Schematic representation of inter- and intra- molecular interactions.

to relaxation phenomena in nuclear spin systems, which are affected by various timescales and demonstrate intricate reciprocal relationships. The secular approximation holds when the dynamics of nuclear spins are predominantly governed by a robust interaction with the external magnetic field, rendering some internal spin interactions insignificant. This approximation is generally effective in most scenarios, except for cases involving quadrupolar spin interactions, which may necessitate a more detailed examination. In such instances, the secular approximation allows for substituting the term $\mathcal{H}^{\text{int}}$ with a simplified version, $\mathcal{H}^{\text{int0}}$.

Motional averaging takes place when molecules are in rapid motion, causing fluctuations in interaction terms over time; consequently, the previous Hamiltonian, $\mathcal{H}^{\text{int0}}$, can be represented by an averaged value, $\bar{\mathcal{H}}^{\text{int0}}$. This method is usually appropriate for gases and liquids where molecular motion is not overly sluggish.

There are three key types of molecular motion to consider: *translational motion*, *rotational motion*, and *internal molecular movements*. The impact of these motions on spin interactions varies depending on whether they involve nuclei within the same molecule (intramolecular) or between different molecules (intermolecular) (see Figure 1.12 for a schematic representation).

In various solids, gases, and liquids, the orientation of molecules is not static and varies over time. This temporal dependence allows for the application of the ergodic hypothesis, which states that the average orientation of a molecule over a long period can be considered equivalent to the average orientation of a collection of molecules at a single moment.

In gaseous states, both rotational and translational motions (diffusion) occur at very high velocities. Consequently, intramolecular spin interactions tend to average out to their isotropic values, while intermolecular interactions are minimized to the point of being negligible.

Conversely, in liquids, diffusion happens at a considerably slower rate compared to gases, influenced by factors such as molecular size and fluid viscosity. In isotropic liquids, intramolecular spin interactions are averaged to isotropic values due to the rotational motion of the molecules. Short-range intermolecular interactions -those occurring within the spatial region where a molecule diffuses before experiencing NMR relaxation- average out to zero. However, long-range intermolecular interactions between molecules that lie outside their respective diffusion domains remain unaffected by this averaging process. In anisotropic liquids, the motionally averaged spin interactions are contingent upon the orientation of the liquid's director (the alignment direction of the liquid crystal) relative to the magnetic field. The rapid motion of molecules effectively diminishes short-range intermolecular interactions while preserving long-range interactions. Additionally, the rotational motion of molecules can lead to intramolecular spin interactions that differ from isotropic averages.

In solids, atomic movement is typically highly restricted, resulting in an internal spin Hamiltonian significantly influenced by the solid's orientation concerning the magnetic field. As a result, there is limited

averaging of internal spin interactions, allowing both intramolecular and intermolecular interactions to remain relevant.

For spins-1/2, generally, *dipole-dipole>CSA>spin-rotation*. The chemical shift anisotropy (CSA) becomes significant compared to dipole-dipole interactions at higher magnetic fields. The spin–rotation mechanism may also be important for small molecules in non-viscous liquids or gases.

While, for nuclei with spins greater than 1/2, *quadrupole≫dipole-dipole>CSA>spin-rotation*. At low magnetic fields, dipole-dipole and quadrupole interactions are typically the most significant contributors to relaxation processes.

1.2.3 The autocorrelation function

The mathematical description of the NMR relaxation induced by the magnetic field variations requires the introduction of some terms.

The rapidity of the fluctuations of the polarizing field $\boldsymbol{B}_0$ is described by the *autocorrelation function*, $G(t)$, defined as

$$G(\tau) = \langle B(t) \cdot B(t+\tau) \rangle \tag{1.17}$$

These parentheses can refer to an average over a long time for a single spin, or an average over the ensemble of n spins at a given time (due to the ergodic hypothesis). The time interval τ is fixed, and $B(t)$ term represents the molecular position and/or orientation expressed in terms of polar coordinates.

Two fundamental properties apply to stochastic fluctuations:

- *Stationarity*. The autocorrelation function is invariant under shifts in the time origin, i.e., it varies with the interval τ but remains unaffected by the absolute time t. As a result, for simplification purposes, the origin is frequently set to zero, i.e., $G(\tau) = \langle B(0) \cdot B(\tau) \rangle$.

- *Time-reversal*. The autocorrelation function remains unchanged when the sign of τ is reversed. In other words, $G(\tau)$ are even functions of τ, i.e., $G(\tau) = G(-\tau) = G(|\tau|)$.

When the interval τ is relatively short compared to the timescale of fluctuations, the values of the field at the two-time points are quite similar, i.e., $G(\tau) \cong \langle B^2(0) \rangle$. In contrast, when τ is long relative to the timescale of fluctuations, the system effectively loses its 'memory'. As a result, no consistent relationship exists between the function values, leading the ensemble average to approach zero. Thus, generally, $G(\tau)$ tends to be large for small values of τ and tends to zero for large values of τ.

Often, the autocorrelation function takes a simple exponential form

$$G(\tau) = \langle B^2 \rangle \, e^{-\frac{\tau}{\tau_c}}$$

where τ_c is the *characteristic time* of the correlation decay defined by

$$\tau_c = \frac{1}{G(0) - G(\infty)} \int_0^\infty [G(t) - G(\infty)] \, dt$$

Correlation time is the duration over which fluctuations in a field are observed. In other words, it indicates how long it takes for these fluctuations to reverse their sign, i.e., extended fluctuations lead to large values of τ_c, while shorter fluctuations lead to short values of τ_c. From (1.17)

$$\tau_c = \int_0^\infty e^{-\frac{t}{\tau_c}} \, dt$$

To define $G(\tau)$ in terms of molecular dynamics, let's introduce a function $f(\boldsymbol{r}(t))$, with $\boldsymbol{r}$ a stochastic position vector. f is real ($f^\star = f$, where $f^\star$ is the complex conjugate of f), and stationary, i.e., it is invariant

under transformations of the time variable. Hence, the autocorrelation function is defined as

$$G(\tau) = \langle f(0) \cdot f(\tau) \rangle = \int_{\mathbb{R}^3} \int_{\mathbb{R}^3_0} f(\boldsymbol{r}) f(\boldsymbol{r}_0) P(\boldsymbol{r}, \boldsymbol{r}_0, \tau) P_0(\boldsymbol{r}_0) d^3\boldsymbol{r}_0 d^3\boldsymbol{r}$$

where $P(\boldsymbol{r}, \boldsymbol{r}_0, \tau)$ represents the conditional probability that a molecule is at the position $\boldsymbol{r}$ at time τ starting from $\boldsymbol{r}_0$ at time 0, while $P(\boldsymbol{r}_0)$ represents the probability to have the molecule in $\boldsymbol{r}_0$ at $t_0 = 0$.

In general terms, this physical quantity is related to $F^{(k)}$ from (1.2.2), thus, it is related to the order k, $G_k(t)$. In the case of the spatial harmonics function $Y_{2,m}$, there will be three correlation functions, i.e., $G_0(t)$, $G_1(t)$, $G_2(t)$. Considering rotational or translational diffusion, the probability $P(\boldsymbol{r}, \boldsymbol{r}_0, \tau)$ can be derived from the classical diffusion equation.

Finally, considering molecules in a confined environment, and using the normalized correlation function $\mathcal{G}(\tau) = \frac{\langle f(0) \cdot f(\tau) \rangle}{\langle |f|^2 \rangle}$, distinguishing between the individual reduced correlation functions is practically not necessary in most cases, i.e., $G_0(t) = G_1(t) = G_2(t) = G(t)$. Hence, with $G(0) = 1$, and assuming $G(\infty) = 0$

$$\tau_c = \int_0^\infty \mathcal{G}(t) dt$$

It is now necessary to introduce the last term, the *spectral density*, $J(\omega)$. This term is obtained by the Fourier transform of the autocorrelation function $G(t)$

$$J(\omega) = \int_{-\infty}^{+\infty} G(t) e^{-i\omega t} dt$$

Spectral density measures the molecular motion at the specific frequency required to produce the relaxation. It is important to note that the field will influence the spins only if it oscillates at the corresponding Larmor frequency. Hence, The function $J(\omega)$ characterizes field fluctuations resulting from thermal molecular motions in the frequency domain. It represents the conjugate counterpart to the correlation function in the time domain.

Given that $G(t)$ is real and even, it follows that also $J(\omega)$ will be real and even ($J(-\omega) = J(\omega)$)

$$J(\omega) = \int_{-\infty}^{+\infty} G(t) e^{-i\omega t} dt = 2 \int_0^\infty G(t) \cos(\omega t) dt$$

From the normalized autocorrelation function $\mathcal{G}(t)$, the normalized spectral density function is derived

$$\mathcal{J}(\omega) = \int_{-\infty}^{+\infty} \mathcal{G}(t) e^{-i\omega t} dt$$

This quantity satisfies the two following properties:

$$\mathcal{J}(0) = \int_{-\infty}^{+\infty} \mathcal{G}(t) dt = 2\tau_c$$

$$\frac{1}{2\pi} \int_{-\infty}^{+\infty} \mathcal{J}(\omega) d\omega = \mathcal{G}(0) = 1$$

By utilizing the quantities defined above, one can establish that the frequency dependence of the spin-lattice relaxation rate $R_1(\omega)$ specifically in the context of intradipolar interactions among like spins, adheres to a relationship similar to that described by Bloch, Wangsness, and Redfield theory[54, 55]

$$R_1(\omega) = \frac{1}{T_1(\omega)} \propto \mathcal{J}(\omega) + 4\mathcal{J}(2\omega) \tag{1.18}$$

Eventually, it is possible to derive a similar result for $R_2(\omega)$

$$R_2(\omega) = \frac{1}{T_2(\omega)} \propto 3\mathcal{J}(0) + 5\mathcal{J}(\omega) + 2\mathcal{J}(2\omega)$$

1.3 Physical Models to interpret FFC Data

In studying nuclear magnetic relaxation dispersion profiles acquired through field-cycling techniques, various mathematical models serve as essential tools for characterizing the molecular dynamics of liquids, both in bulk systems and under confinement conditions. Several models have been developed and proposed, and they enable the interpretation of relaxation phenomena across a range of Larmor frequencies, typically $\nu_0 \in [10 - 4000] \cdot 10^3$ kHz, subject to the limitations of the instrumentation. To accurately capture these dynamics, certain approximations must be introduced. The following section discusses the most well-known models in this field, including the model subject of study of this PhD thesis, i.e., the *Model-Free* (MF).

1.3.1 Spin Dynamics in Bulk Liquids

The behaviour of spin-bearing molecules in bulk liquids is shaped by intermolecular and intramolecular interactions, which operate on different timescales. Intermolecular interactions exhibit slower fluctuations compared to their intramolecular counterparts. Specifically, intermolecular couplings are influenced by translational Brownian motion, which occurs over distances that exceed the size of the molecules. In contrast, intramolecular interactions are governed by the rapid process of molecular rotational diffusion.

Dipole-dipole interactions can occur within a single molecule (intramolecular) and between different molecules (intermolecular), whereas quadrupolar interactions are restricted to intramolecular scenarios. The influence of intermolecular dipole-dipole interactions on longitudinal relaxation is significant primarily under conditions of anisotropic molecular dynamics, where these interactions manifest at lower frequencies. However, their impact is considerably reduced in isotropic environments.

When considering that rotational and translational diffusion are statistically independent, the total longitudinal relaxation rate can be represented as the sum of contributions from intermolecular and intramolecular dipole-dipole interactions among spins that may be categorized as "like" or "unlike." For diamagnetic samples, this relationship can be articulated mathematically as:

$$\frac{1}{T_1(\omega)} = \frac{1}{T_1^{\text{inter}}(\omega)} + \frac{1}{T_1^{\text{intra}}(\omega)}$$

The degree to which these relaxation rates contribute to the overall rate depends on various system-specific factors, including composition, molecular dynamics, and the frequency range utilized in FFC-NMR experiments. In cases where quadrupolar couplings are present, these interactions can influence relaxation across all frequencies. Due to their stronger nature compared to dipolar couplings, intramolecular quadrupolar interactions often dominate longitudinal relaxation when paramagnetic particles are absent:

$$\frac{1}{T_1(\omega)} \simeq \frac{1}{T_1^{\text{inter}}(\omega)}$$

To accurately capture spin dynamics, it is essential to develop a conceptual model that reflects the processes and interactions occurring within the pore volume. This model accounts for interactions between spins as well as between spins and pore surfaces, facilitating the derivation and calculation of the autocorrelation function. The spectral density function $J(\omega)$ can then be obtained through the Fourier transform of $G(t)$, linking it to relaxation dispersion curves for both T_1 and T_2 relaxation times.

In this context, all relevant interactions can be described using fluctuating spherical coordinates that represent variations in dipole-dipole vectors or electric field gradients. For instance, dipolar interactions can be characterized by spherical coordinates defined by internuclear distance and azimuthal and polar angles, $\boldsymbol{r} = \boldsymbol{r}(t)$, $\varphi = \varphi(t)$, and $\theta = \theta(t)$ respectively. In contrast, for quadrupolar couplings, only azimuthal and polar angles are necessary, with changes in internuclear distance considered negligible.

1.3.2 BPP Theory

One of the most well-known theories related to the relaxation processes was proposed in 1948 by Nicolaas Bloembergen, Edward Mills Purcell, and Robert Pound, and it is known as *BPP theory*. This theory describes the relaxation process in scenarios characterized by a high-intensity magnetic field (known as a 'weak collision' regime) where isotropic rotational molecules diffusion, and intramolecular dipole-dipole interactions within two-spin $\frac{1}{2}$ systems which exhibit similar spin properties (i.e., $I = S$, and $\gamma_I = \gamma_S$) occur in bulk liquids.

Considering the case of isotropic rotational diffusion in bulk liquids, the reduced correlation function is a mono-exponential function, i.e., $\mathcal{G}(t) = e^{-\frac{t}{\tau_c}}$, (with $G(0) = 1$). Therefore, the correspondent spectral density function is defined as $\mathcal{J}(\omega) = \int_{-\infty}^{+\infty} \mathcal{G}(t) e^{-i\omega t} dt = \frac{2\tau_c}{1+(\omega\tau_c)^2}$, presenting a Lorentzian form.

Hence, in the BPP model, the relaxation rates expressed as a function of $J(\omega)$ are defined as [56]

$$R_1(\omega) = \frac{1}{T_1(\omega_0)} = \frac{1}{5}\left(\frac{\mu_0}{4\pi}\right)^2 \frac{\gamma^4\hbar^2}{r^6} I\,(I+1)\,[J(\omega_0) + 4J(2\omega_0)]$$

$$R_2(\omega) = \frac{1}{T_2(\omega_0)} = \frac{1}{10}\left(\frac{\mu_0}{4\pi}\right)^2 \frac{\gamma^4\hbar^2}{r^6} I\,(I+1)\,[3J(0) + 5J(\omega_0) + 2J(2\omega_0)]$$

and by substituting the $\mathcal{J}(\omega)$ expression, for the dipolar coupling of two spins $\frac{1}{2}$ systems:

$$R_1(\omega) = \frac{1}{T_1(\omega_0)} = \frac{3}{10}\left(\frac{\mu_0}{4\pi}\right)^2 \frac{\gamma^4\hbar^2}{r^6}\left[\frac{\tau_c}{1+\omega_0^2\tau_c^2} + \frac{4\tau_c}{1+(2\omega_o\tau_c)^2}\right]$$

$$R_2(\omega) = \frac{1}{T_2(\omega_0)} = \frac{3}{20}\left(\frac{\mu_0}{4\pi}\right)^2 \frac{\gamma^4\hbar^2}{r^6}\left[3\tau_c + \frac{5\tau_c}{1+\omega_0^2\tau_c^2} + \frac{2\tau_c}{1+(2\omega_o\tau_c)^2}\right]$$

where $\omega_0 = \gamma B_0$ is the Larmor angular frequency, γ is the gyromagnetic ratio, $\hbar$ is the reduced Planck constant, i.e., $\hbar = h/2\pi$, and μ_0 is the magnetic field constant.

In case of extreme narrowing conditions, e.g., in non-viscous liquids, with $\omega_0\tau_c \ll 1$, the rates have the same value.

$$\frac{1}{T_1} = \frac{1}{T_2} = \frac{3}{2}\left(\frac{\mu_0}{4\pi}\right)^2 \frac{\gamma^4\hbar^2}{r^6}\tau_c$$

The BPP model, assuming the mono-exponential correlation function, represents ideal cases, allowing one to describe and agree to easy experimental situations, e.g., experiments with pure substances.

1.3.3 Quadrupolar coupling in NMRD profiles

In specific systems, interesting behaviours arise when dipolar I-spins interact with quadrupolar S-spins, particularly when the S-spins undergo additional quadrupolar interactions with electric-field gradients produced by nearby host molecules. When a quadrupolar energy level of the S-spin nuclei coincides with the Larmor frequency of the I-spins, this alignment can lead to the appearance of distinct quadrupolar features in the NMRD profile. These features are characterized by dips, and peaks in the $T_1(\omega)$, and in the $R_1(\omega)$ dispersion curves respectively.

As the polarizing field approaches the low-field limit, i.e., $B_0 \to 0$, the electric-field gradient becomes the dominant quantization field. In cases where molecular motion is constrained, the quadrupolar interaction of the S-spins leads to zero-field splitting of their energy levels. Conversely, in the high-field limit, where B_0 is the quantization field, and the Zeeman splitting of the S-spin energy levels prevail. Under both limits, the S-spins maintain thermal equilibrium, unaffected by fluctuations in the I-spin states.

In general, the gyromagnetic ratio of dipolar I-spins is considerably higher than that of quadrupolar S-spins, i.e., $|\gamma_I| \gg |\gamma_S|$. Under conditions where the B_0 field is relatively weak and the resonance of the S-spin remains within the low-field regime, the quadrupolar interaction becomes predominant, establishing the main energy level splitting. In this context, there exists a possibility for the resonance frequency of the I-spin, which is determined by Zeeman interaction, to align with that of the S-spin.

The intersection of resonant frequencies occurs when the resonance frequency of quadrupolar S-spins in a low-field environment aligns with that of the I-spin, $\omega_S = \omega_I = \gamma_I B_0$. This synchronization triggers a zero-quantum "flip-flop" mechanism, which results in a decreased spectral density term that peaks at the difference between the frequencies, $\omega_I - \omega_S$, with the dipolar correlation time, τ_{dip} is subject to various influences that introduce fluctuations in the dipole-dipole interactions between I-spins and S-spins

$$J(\omega_I - \omega_S) \propto \frac{\tau_{\text{dip}}}{1 + (\omega_I - \omega_S)^2 \tau_{\text{dip}}^2}$$

This phenomenon is reflected as a distinct peak in the relaxation rate curve across the magnetic field, typically referred to as a quadrupole peak.

1.3.4 Spin Dynamics in Confined Fluids

The magnetic field dependence of the longitudinal relaxation rate offers a precise means to test theoretical models that elucidate the behaviour of confined liquids, where the influence of reduced dimensionality alters molecular interactions significantly. Developing comprehensive models that link the molecular dynamics of fluids in porous networks to the frequency-dependent behaviour of NMRD relaxometry enables probing fluid dynamics at the nanoscale with unprecedented insight.

The BPP theory successfully characterized bulk glycerin through the assumption of an exponentially decaying $G(t)$, reflecting molecular tumbling in a bulk phase. However, when investigating fluids in confined environments -such as those occupying porous matrices or surrounding high surface area particles- the interaction landscape becomes considerably more intricate, as molecular dynamics are now mediated by surface and interfacial interactions, deviating from bulk behaviour.

Several analytical models have emerged to address this complexity in Magnetic Resonance of Porous Media (MRPM) and complex fluids. These models reduce the dimensional intricacies of confined fluid systems while retaining critical physical phenomena essential for the accurate interpretation of NMRD profiles. Notably, Levitz et al. advanced a surface-mediated diffusion model tailored to biological contexts, focusing on the impact of surface dynamics on relaxation [57]. In parallel, Kimmich and co-workers, inspired by Bychuk and O'Shaughnessy's foundational 1995 work, developed the bulk-mediated surface diffusion (BMSD) model [1]. This framework posits a sequential mechanism whereby molecules desorb from a surface, undergo diffusion in a bulk-like interfacial layer, and subsequently re-adsorb onto the surface, thereby capturing the interplay between near-surface and bulk-like diffusion.

Among the more comprehensive frameworks, Korb and colleagues developed a generalized model in the early 1990s for describing relaxation dispersion in confined fluids [58, 59]. Their model emphasizes the dominant relaxation pathways for hydrogen-containing molecules through heteronuclear dipolar interactions with paramagnetic ions (e.g., Fe^{3+} or Mn^{2+}) distributed on pore surfaces. Primarily applicable at low frequencies and over extended timescales, these models are often categorized within quasi-two-dimensional (Q2D) frameworks, relying on three distinct correlation times to represent the complex relaxation dynamics of fluids in geometrically constrained environments.

Expanding on the Q2D approach, Faux et al. introduced the *3τ model* in the early 2000s [60]. This model provides a refined description of diffusion correlation times across both thin surface-adjacent layers and the bulk phase, accommodating the nuanced relaxation dynamics at the fluid-solid interface. By integrating

numerical estimation techniques, the 3τ model extends the applicability of Q2D frameworks, offering a robust approach for simulating confined fluid behaviour across diverse porous architectures.

Porous Media: Effects of Surface Impurities

Korb and collaborators categorized high-surface-area systems into two principal types: proton-rich solid phases, such as biological macromolecules and engineering polymers, and proton-poor solid phases, including microporous glasses, zeolites, plaster, cement, and various clay minerals. In the proton-rich class, the magnetic field dependence of the relaxation rate is largely governed by cross-relaxation processes between protons in the liquid phase and those embedded within the solid matrix. By contrast, in proton-poor materials, the relaxation of liquid spins is predominantly influenced by alternative mechanisms, especially those associated with translational diffusion of the fluid near the solid surfaces [61].

In high-surface-area media containing paramagnetic centres (e.g., certain cement and geological materials), the presence of paramagnetic ions, such as manganese (Mn^{2+}) and iron (Fe^{3+}), can profoundly affect the behaviour of NMR relaxation. These ions possess large magnetic moments, producing localized dipolar fields that interact with nearby diffusing spins in the fluid. The substantial magnetic moments of these paramagnetic centres significantly dominate the ^1H longitudinal relaxation process at low magnetic field strengths. Additionally, theoretical models exist for cases where no paramagnetic contaminants are present, providing a comprehensive approach to analysing the longitudinal relaxation in confined fluid systems.

Confined fluid relaxation times are generally characterized by two distinct limiting behaviours: diffusion-limited (slow-diffusion) relaxation, where the relaxation is governed by bulk diffusion, and surface-limited (fast-diffusion) relaxation, where the interactions with the surface prevail. In the surface-limited regime, the total proton relaxation rate $1/T_1$ is described as a linear combination of the surface relaxation rate $1/T_{1,s}$ and the bulk relaxation rate $1/T_{1,b}$. Here, the first one occurs within a thin layer of fluid, approximately λ thick, adjacent to the solid surface, where adsorbed protons undergo diffusion before eventually desorbing and diffusing into the bulk phase. Notably, in the frequency ranges used for porous media analysis via FFC-NMR, the second one remains independent of the Larmor angular frequency.

To assess the impact of the surface distribution of paramagnetic sites on a proton located at a maximum distance λ from the interface of a solid grain (typically several microns in size), the surface of the pore can be approximated as flat, given that the curvature of the pore wall is considerably larger than λ. $T_{1,s}$ can be expressed as a composite contribution from two processes: the relaxation contribution $1/T_{1,2D}$ of protons diffusing near immobilized paramagnetic centres, and the relaxation rate $1/T_{1,p}$ of protons trapped in the ligand fields of these centres

$$\frac{1}{T_1(\omega)} = \frac{1}{T_{1,b}} + \frac{N_S}{N_b}\frac{1}{T_{1,2D}(\omega)} + \frac{N_P}{N_b}\frac{1}{T_{1,p}(\omega)}$$

In this model, N_S denotes the number of diffusing molecules within the transient layer λ near the pore surface, N_P the number of liquid molecules associated with paramagnetic impurities on the surface, and N_b the population of molecules in the bulk liquid phase.

Furthermore, the nature of the saturating fluid significantly impacts relaxation behaviour, with fluids classified as either protic or aprotic [62]. Protic liquids, which can exchange protons, allow mobile protons to be temporarily trapped within the ligand fields of paramagnetic sites on the pore surface. Aprotic liquids, lacking proton exchange capacity, restrict the movement of protons, preventing such trapping interactions with surface paramagnetic sites.

For the protic liquids, the protons (I) might be trapped in the paramagnetic ion (S) field for a time longer than the bulk or surface correlation times. The value for the $1/T_{1,p}$ rate is given by [59]

$$\frac{1}{T_{1,p}(\omega_I)} = \frac{2}{15}\frac{\gamma_I^2\gamma_S^2\hbar^2 S(S+1)}{D_{\min}^6} T_{1,\mathrm{elec}}(\omega_S)\left[\frac{3}{1+\omega_I^2 T_{1,\mathrm{elec}}^2(\omega_S)} + \frac{7}{1+\omega_S^2 T_{1,\mathrm{elec}}^2(\omega_S)}\right]$$

where $D_{\min}$ represents the distance of minimal approach between the paramagnetic site and the proton, ω_I, and ω_S are the proton and the electronic angular frequency respectively. The electronic correlation time is given by $\frac{1}{T_{1,\text{elec}}} = H_S^2 \tau_\nu \left[\frac{1}{1+\omega_S^2\tau_\nu^2} + \frac{4}{1+4\omega_S^2\tau_\nu^2}\right]$, with H_S^2 the intensity of electron-spin fluctuations, and τ_ν the correlation time for the electron-lattice fluctuating interaction.

Finally, for the case of the aprotic liquid, the contribution given by $T_{1,p}$ can be neglected because the protons tend to not get trapped in the paramagnetic ion field. The longitudinal relaxation rate can be modelled [62] at low frequencies as

$$\frac{1}{T_1(\omega_I)} = \frac{1}{T_{1,b}} + \left(\frac{\mu_0}{4\pi}\right)^2 \frac{\pi S_P \rho_{\text{liq}} \sigma_S \gamma_I^2 \gamma_S^2 \hbar^2 S(S+1)}{\lambda D_{\min}^2 (1+x^2)} \tau_m \left[3\ln\left(\frac{1+\omega_I^2\tau_m^2}{\left(\frac{\tau_m}{\tau_S}\right)^2 + \omega_I^2\tau_c^2}\right) + 7\ln\left(\frac{1+\omega_S^2\tau_m^2}{\left(\frac{\tau_m}{\tau_S}\right)^2 + \omega_S^2\tau_c^2}\right)\right]$$

where τ_S is the surface residence time, which indicates the duration that proton-bearing molecules spend at a pore surface. This time reflects how long these molecules diffuse within the thin surface layer, λ, and serves as an indication of the duration for which protons remain associated with paramagnetic impurities. On the other hand, τ_m denotes the translational correlation time, which is linked to the individual molecular movements occurring near the surface. The parameter x serves to account for the variable distance, $D_{\min}$, representing the minimal approach between proton spins and impurity spins in relation to the diameter, D, of the liquid molecules. Conclusively, $\sigma_S = \eta_S \rho_S \xi$ represents the surface density of a small quantity of fixed paramagnetic species with S-spins that are evenly distributed across the pore surfaces. Here, ρ_S denotes the density of the solid matrix, η_S indicates the volume concentration of paramagnetic ions, and ξ refers to the thin layer of paramagnetic ions.

3τ Model

An alternative to the Korb model to analyse the NMRD profiles acquired from porous media saturated with ^1H fluids has been proposed in the early 2000 by Faux and McDonald [60], and it is known as 3τ model.

The composition of the porous media and the intricate internal structure makes very complex the dynamic of the fluids, and therefore to model it. If the porous material has paramagnetic ions, the proton-proton interactions become negligible and the relaxation mechanism is predominantly governed by dipolar interactions between proton spins and the paramagnetic ions; otherwise, in the absence of paramagnetic ions, the relaxation depends on the combination of interactions between pairs of protons in relative motion.

The 3τ model is based on the idea of defining the pore geometry as a quasi-two-dimensional (Q2D) system with locally flat surfaces containing a bulk liquid of thickness h [6, 63], and defining three correlation times:

- τ_b is the bulk fluid correlation diffusion time constant related to the water diffusion coefficient $D_b = \sigma^2/(6\tau_b)$. The value of σ is $0.27 \cdot 10^{-9}$ m, which roughly represents the intermolecular distance between water molecules.

- τ_ℓ is the diffusion correlation time for the fluid slowly moving parallel at the pore surface, related to the diffusion coefficient $D_\ell = \sigma^2/(6\tau_\ell)$.

- τ_d is the desorption time, which represents the time a water molecule stays on the surface before the desorption, taking the rate of spins desorption as an exponential decreasing function, i.e., $e^{-\frac{t}{\tau_d}}$.

The model provides longitudinal relaxation rates for each interaction.

For the interaction between the layer of spins of the paramagnetic impurity in the crystal (σ) and the surface (ℓ) the relaxation rate is $\frac{1}{T_{1,\sigma\ell}}$, while between σ and the bulk (b) fluids, the relaxation rate is $\frac{1}{T_{1,\sigma b}}$.

Considering the surface water moves very slowly compared to the bulk fluid, hence, it can be considered effectively immobile due to its extremely slow movement, i.e., $\tau_\ell \gg \tau_b$, the relaxation rate for the interaction between bulk spins and surface fluid is defined as $\frac{1}{T_{1,\ell b}}$.

For the pore surface layer, $\frac{1}{T_{1,\ell\ell}}$ represents the relaxation rate for the layer-layer spin interactions, while $\frac{1}{T_{1,\ell\ell}}$ is the relaxation rate for the bulk-bulk spins interactions.

Finally, the relaxation rate due to paramagnetic ions, e.g., manganese, dispersed in water is defined as $\frac{1}{T_{Mn}}$.

Hence, taking into account all these contributions, and ignoring the last one, for the case of systems containing paramagnetic impurities, the measured relaxation rate can be modelled as

$$R_1 = \frac{n_\ell}{n_\ell + n_b}\frac{1}{T_{1,\sigma\ell}} + \left(1 - \frac{n_\ell}{n_\ell + n_b}\right)\frac{1}{T_{1,\sigma b}}$$

where $\frac{n_\ell}{n_\ell+n_b}$ defines the amount of spins in the surface layer volume. While, the relaxation rate in the case of systems not containing paramagnetic impurities, is

$$R_1 = \frac{n_\ell}{n_\ell + n_b}\left(\frac{1}{T_{1,\ell b}} + \frac{1}{T_{1,\ell\ell}}\right) + \left(1 - \frac{n_\ell}{n_\ell + n_b}\right)\left(\frac{1}{T_{1,b\ell}} + \frac{1}{T_{1,bb}}\right)$$

where $\frac{1}{T_{1,b\ell}} = \frac{N_b}{N_\ell}\frac{1}{T_{1,\ell b}}$, with N_b, and N_ℓ the spin volume density of the bulk fluids and surface layer respectively.

To cover the full-time domain, the analytical expression of the correlation time function in this model is numerically computed via Monte Carlo simulations. The correspondent spectral density function is obtained by numerical integration. Moreover, the parameters of the Monte Carlo simulations are derived through molecular dynamics simulations of bulk water and water in Q2D systems. This means that the model requires a single formulation of the $G(t)$ for each interaction described before, i.e., $G_{\ell b}(t)$, $G_{\ell\ell}(t)$, $G_{bb}(t)$, $G_{\sigma b}(t)$, and $G_{\sigma\ell}(t)$.

Firstly, the probability density function $P(\boldsymbol{r}_0 \cap \boldsymbol{r}, t)$ is defined. This defines the probability that the distance between two spins at $t = 0$ is $\boldsymbol{r}_0$, and then at time t it becomes $\boldsymbol{r}$. This probability depends on a set of parameters related to the various interactions between pairs of spins, i.e., the correlation times. Then, recalling from 1.2.2, the spherical harmonic functions of degree 2 can be expressed through cylindrical coordinates, i.e, $Y_{2,m}(x, y, z)$[5]. Therefore, the general expression for the correlation function is[63]

$$G(t) = \frac{4\pi}{5}\int_{\mathbb{R}^3}\int_{\mathbb{R}^3_0}\left[\sum_{m=-2}^{2}\frac{Y_{2,m}(x_0, y_0, z_0)Y^*_{2,m}(x, y, z)}{\boldsymbol{r}_0^3\boldsymbol{r}^3}\right]P(\boldsymbol{r}_0 \cap \boldsymbol{r}, t)d^3\boldsymbol{r}_0 d^3\boldsymbol{r}$$

and the spectral density function is obtained by the Fourier transform of $G(t)$, i.e., $J(\omega) = 2\int_0^\infty G(t)\cos(\omega t)dt$.

The longitudinal relaxation rate for the systems with and without paramagnetic ions respectively are defined as [63]

$$R_1^{\text{param}}(\omega) = \frac{1}{3}\left(\frac{\mu_0}{4\pi}\right)^2\gamma_I^2\gamma_S^2\hbar^2 S(S+1)\left[7J(\omega_S) + 3J(\omega)\right]$$

$$R_1(\omega) = \frac{1}{5}\left(\frac{\mu_0}{4\pi}\right)^2\gamma_I^4\hbar^2 I(I+1)\left[J(\omega) + 4J(\omega)\right]$$

where γ_S and γ_I are the impurities and the proton gyromagnetic ratios respectively, while $\omega = 2\pi\nu$ represents the proton Larmor angular frequency, and $\omega_S = 658.2\omega$ represents the electron Larmor angular frequency of the impurity spins (typically Mn^{2+}, and Fe^{3+}). $I = \frac{1}{2}$, and generally $S = \frac{5}{2}$.

The 3τ model was initially developed for porous materials with flat pores of depth h. It has been shown that when pore thickness exceeds $5 \cdot 10^{-9}$ m, the dispersion of longitudinal relaxation time becomes

[5] with Y^* the complex conjugate.

unaffected by the pore's thickness [64]. Consequently, this model can be adapted to systems where the fluid is contained within pores with at least two dimensions larger than $5 \cdot 10^{-9}$ m. However, despite its theoretical potential, practical application remains challenging. This difficulty arises mainly from the complex calculations necessary to derive the theoretical R_1 curve from the model, which complicates its alignment with experimental data. Faux et al. also developed a code based on this model, which allows users to perform least squares fitting on data (loading pre-calculated data sets based on the model for a pre-defined grid of values for τ_b, τ_ℓ, and τ_d) [65].

1.3.5 Model-Free

The model-free approach, a breakthrough in nuclear magnetic resonance dispersion (NMRD) analysis, was first developed in the early 1980s by scientists including Lipari, Szabo, King, and Jardetzky [66]. This methodology, which allows for detailed insights into molecular motion without the need for explicit modeling, was later advanced by Halle and collaborators [25], who expanded its utility for both experimental and theoretical applications, leading to a more rigorous interpretation of relaxation dynamics.

This technique has proven especially powerful in cases where molecular dynamics are complex or heterogeneous, as it allows for the study of a wide variety of motional modes without the risk of oversimplification inherent in rigid interpretative models.

In situations where experimental NMRD data spans the full frequency range -covering both the low- and high-frequency plateaus- it becomes possible to extract dynamic parameters that describe molecular motions independently of a specific molecular model. This feature lies at the heart of the model-free approach, as it enables the direct extraction of physical information from data without the assumptions that typically constrain model-based interpretations.

The rate of nuclear spin relaxation becomes dependent on frequency when the motional frequencies that affect the spin-lattice coupling are comparable to the energy level separations within the spin system[46]. In the regime known as motional narrowing, the influence of a fluctuating classical lattice variable $V(t)$ on the behaviour of spin relaxation can be effectively described through the time correlation function

$$G(\tau) = \langle V(t) \cdot V(t+\tau) \rangle$$

The function $G(\tau)$ is a real-valued quantity. In dynamic models that adhere to the detailed balance condition, $G(t)$ exhibits invariance under time reversal, meaning its properties remain consistent even when the direction of time is reversed, i.e., $G(\tau) = G(-\tau)$.

The complex-valued spectral density function is defined as

$$J(\omega) = \int_{-\infty}^{\infty} G(\tau) e^{-i\omega\tau} d\tau = 2 \int_{0}^{+\infty} G(\tau) \cos(\omega\tau) d\tau$$

It's worth noting that

$$J(0) = 2 \int_{0}^{\infty} G(\tau) d\tau \tag{1.19}$$

hence $G(\tau)$ must be an integrable function, i.e., $G(\tau) \underset{\tau \to \infty}{\to} 0$.

Let us consider a simple dynamic case, as rotational diffusion of spherical-top molecules in an isotropic environment. In this case, $G(\tau) = G(0) e^{-\frac{|\tau|}{\tau_c}}$, with τ_c describing the $V(t)$ fluctuations time scale, and $G(0) = \langle V^2 \rangle$ represents the mean square amplitude of the fluctuations. Then, the spectral density function will have a Lorentzian form, only depending on two parameters, $G(0)$, and τ_c:

$$J(\omega) = G(0) \frac{2\tau_c}{1 + (\omega\tau_c)^2}$$

From (1.18), R_1 is proportional to J minus a constant, K, which depends on the specific spins coupling

$$R_1(\omega) = K\left[J(\omega) + 4J(2\omega)\right]$$

NMRD profiles typically exhibit a broader frequency range, characterized by a *stretched* dispersion, compared to what would be expected from a purely Lorentzian spectral density. While a stretched dispersion provides additional insights into the system, a Lorentzian dispersion can be fully described using just two parameters.

The first one is represented by the integral of the dispersion profile, which represents the mean-square fluctuations of the lattice variable $V(t)$. Hence, taking into account that G is real and satisfies the time reversal property,

$$\int_0^\infty J(\omega)d\omega = \frac{1}{2}\int_{-\infty}^\infty J(\omega)d\omega = \frac{1}{2}\int_{-\infty}^\infty \left(\int_{-\infty}^\infty G(\tau)e^{-i\omega\tau}d\tau\right)d\omega =$$
$$= \frac{1}{2}\int_{-\infty}^\infty G(\tau)\left(\int_{-\infty}^\infty e^{-i\omega\tau}d\omega\right)d\tau = \frac{1}{2}\int_{-\infty}^\infty G(\tau)2\pi\delta(\tau)d\tau$$

where $\delta(\tau)$ is the Dirac function, i.e., $\delta(\tau) = \frac{1}{2\pi}\int_{-\infty}^\infty e^{-i\omega\tau}d\omega$. Hence,

$$\int_0^\infty J(\omega)d\omega = \pi G(0)$$

The second parameter required to describe the Lorentzian dispersion is the mean motion duration related to stretched dispersion, i.e., the integral over the time of the reduced time correlation function. Considering (1.19),

$$\langle\tau_c\rangle = \frac{1}{G(0)}\int_0^\infty G(\tau)d\tau = \frac{1}{2}\frac{J(0)}{G(0)} \tag{1.20}$$

These two quantities can be derived by fitting experimental raw data through a mathematical model capable of physically correctly representing the NMRD profiles. The stretched spectral density can be represented as a sum of N Lorentzian terms

$$J(\omega) = \sum_{n=1}^N c_n \frac{2\tau_n}{1+(\omega\tau_n)^2} \tag{1.21}$$

while the correlation function is represented as a sum of exponentials

$$G(\tau) = \sum_{n=1}^N c_n e^{-\frac{|\tau|}{\tau_n}}$$

Calculating the set of parameters $\{\tau_n, c_n\}$, one can be compute the molecular dynamics quantities $G(0)$, and $\langle\tau_c\rangle$. Starting from the integration of the spectral density function

$$\int_0^\infty J(\omega)d\omega = \sum_{n=1}^N c_n\tau_n \int_0^\infty \frac{2}{1+(\omega\tau_n)^2}d\omega = 2\sum_{n=1}^N c_n\cancel{\tau_n}\left[\frac{1}{\cancel{\tau_n}}\arctan(\omega\tau_n)\right]_0^\infty =$$
$$= 2\sum_{n=1}^N c_n\left[\frac{\pi}{2} - 0\right] = \pi\sum_{n=1}^N c_n = \pi G(0)$$

one obtains

$$G(0) = \sum_{n=1}^N c_n$$

and, setting setting $\omega = 0$ in (1.21), one obtains $J(0) = 2\sum_{n=1}^N c_n\tau_n$, and the (1.20) becomes

$$\langle\tau_c\rangle = \frac{1}{2}\frac{J(0)}{G(0)} = \frac{1}{\cancel{2}}\frac{\cancel{2}\sum_{n=1}^N c_n\tau_n}{\sum_{n=1}^N c_n}$$

The set of parameters $\{c_n, \tau_n\}$ can be fitted following two approaches, i.e., the non-linear discrete approach, or the linear quasi-continuous approach. Mathematically, identifying the model-free parameters starting from the experimental observation represents an ill-conditioned non-linear inverse problem. The necessary theoretical preliminaries related to inverse problems are presented in Chapters 2, and 3.

The two approaches are successively expressed. Specifically, the second approach defines some of the contributions of this PhD thesis and it is comprehensively adapted and described in Chapters 5, and 6.

Non-linear Discrete Model-Free

The non-linear discrete model-free approach finds the set of parameters by solving the classical non-linear least squares problem [53].

Let us consider the set of parameters organized in two separate vectors $\boldsymbol{c} \equiv (c_1, \ldots, c_n)^T$, and $\boldsymbol{\tau} \equiv (\tau_1, \ldots, \tau_n)^T$. The residual vector $\boldsymbol{r}$ as a function of c, and τ is derived as the difference between the NMRD profile experimental measured points (R_1^{exp}), and the NMRD profile from the theoretical fitted model (R_1^{th}):

$$\boldsymbol{r}(\boldsymbol{c}, \boldsymbol{\tau}) \equiv R_1^{\text{exp}}(\omega_i) - R_1^{\text{th}}(\omega_i) \ , \qquad i = 1, \ldots, M$$

Therefore, the constrained non-linear weighted least squares problem to find $\boldsymbol{c} \geq 0$, and $\boldsymbol{\tau} \geq 0$, is

$$\min_{(\boldsymbol{c}, \boldsymbol{\tau}) \geq 0} \sum_{i=1}^{M} w_i \boldsymbol{r}\left(\boldsymbol{c}, \boldsymbol{\tau}\right)^2 \tag{1.22}$$

where $w_i \geq 0$ represents the $i-th$ weight related to the data noise, with $i = 1, \ldots, M$. If the noise is independent of i, $w_i = 1$, otherwise it depends on the noise standard deviation either by the σ_i at the signal point $R_1(w_i)$, i.e., $w_i = 1/\sigma_i^2$, either by using strategies to derive the standard deviation from experimental data, e.g., solving (1.22), with $w_i = 1$, $\forall i$, and then using the fitted dispersion profile R_1^{fit}:

$$w_i = \frac{1}{R_1^{\text{fit}}(i)}$$

where $R_1^{\text{fit}}(i)$ is derived considering a threshold $ErrFit = \sqrt{\sum_{j=1}^{\bar{M}} \frac{(\boldsymbol{r}_{\bar{M}}(j))^2}{\bar{M}}}$, with $\boldsymbol{r}_{\bar{M}}$ contains $\bar{M} \simeq 10^1$ smallest components of the residual vector $R_1(\omega_i) - R_{1i}$, as follows: $R_1^{\text{fit}}(i) = \min\left(|R_{1i}|, ErrFit\right)$, $i = 1, \ldots, M$.

The minimization procedure of (1.22) can be carried out using the *Projected Newton Method* [67], with the Hessian matrix modified as in the Levemberg-Marquardt method [68].

Finally, it is important to find a correct strategy to find how many Lorentzian functions should be used in the fitting procedure. One solution can be to adapt the number of Lorentzian terms, N, through the F-test, utilizing the reduced χ_r^2 merit function defined as follows

$$\chi_r^2 = \frac{1}{M - 2N} \sum_{i=1}^{M} w_i r_i \left(\boldsymbol{c}, \boldsymbol{\tau}\right)^2$$

If the data errors are normally distributed, then the ratio $\chi_r^2(N)/\chi_r^2(N+1)$ follows an F-distribution. One can start setting $N = 1$, and then increasing the number of N until the F-probability satisfies the following: $P(N+1) < P_0$, with $0.8 \leq P_0 \leq 0.9$ cut-off.

Linear Quasi-Continuous Model-Free

The system's microscopic dynamics can be characterized through a normalized distribution of correlation times $f(\tau)$ [26]. The nuclear spin-lattice relaxation rate R_1 is related to this distribution through a Fredholm integral equation of the first kind:

$$R_1(\omega) = \int_0^{\infty} \mathcal{K}(\tau, \omega) f(\tau)\, d\tau \tag{1.23}$$

where $\mathcal{K}(\tau, \omega)$ is the Lorentzian kernel:

$$\mathcal{K}(\tau, \omega) = \tau \left[\frac{1}{1 + (\omega\tau)^2} + \frac{4}{1 + (2\omega\tau)^2} \right] \tag{1.24}$$

For numerical implementation, we discretize the problem using:

- A set of logarithmically spaced correlation times $\{\tau_j\}_{j=1}^{N}$ in the interval $[10^{-4}, 10^{2}]$ μs,
- A set of experimental Larmor frequencies $\{\omega_i\}_{i=1}^{M}$, where typically $M < N$.

This leads to the discrete matrix equation:

$$\mathbf{K}\mathbf{f} = \mathbf{R}_1 \tag{1.25}$$

where $\mathbf{K} \in \mathbb{R}^{M \times N}$ is the kernel matrix with elements

$$\mathbf{K}_{i,j} = \mathcal{K}(\tau_j, \omega_i),$$

and $\mathbf{R}_1 \in \mathbb{R}^{M}$ is the vector of relaxation rates

$$\mathbf{R}_1(i) = R_1(\omega_i), \qquad i = 1, \ldots, M.$$

Due to the ill-conditioned nature of $\mathbf{K}$, various regularization strategies are examined in Chapters 5, 6, and 7.

List of Acronyms and Symbols of the Chapter 1

Acronyms

- **NMR** Nuclear Magnetic Resonance.
- **RF** Radio-Frequency.
- **FFC** Fast Field Cycling.
- **FID** Free Induction Decay.
- **SE** Spin Echo.
- **TE** Echo Time.
- **NMRD** Nuclear Magnetic Relaxation Dispersion.
- **CSA** Chemical Shift Anisotropy.
- **MF** Model-Free.

Symbols

- $\boldsymbol{B}_0$. Polarizing or Zeeman field.
- $\boldsymbol{\mu}$. Nuclear magnetic moment.
- ν_0. Larmor angular frequency.
- $\boldsymbol{M}$. Magnetization vector.
- γ. Gyromagnetic ratio.
- h. Planck's constant.
- $\hbar$. Reduced Planck's constant.
- ω_0. Larmor angular velocity.
- $\boldsymbol{B}_1$. Radio-frequency magnetic field.
- α. Flip angle.
- T_1. Longitudinal or spin-lattice relaxation time.
- T_2. Transversal or spin-spin relaxation time.
- χ_0. Magnetic susceptibility.
- $\mathcal{H}$. Nuclear spin Hamiltonian.
- $G(\tau)$. Autocorrelation function.
- τ_c. Characteristic correlation time.
- $J(\omega)$. Spectral density.
- R_1. Longitudinal Relaxation Rate.

Chapter 2

Regularization and Optimization for Inverse Problems

The motivation for this discussion arises from the need to address the challenges associated with the model-free approach used during PhD research.

This chapter describes the regularization strategies and the optimization methods for inverse problems that are required later to derive the proposed numerical algorithms. Specifically, in the first section of the chapter, Tikhonov regularization for linear and non-linear problems is described in a more general context. The last section discusses the block non-linear Gauss-Seidel method and its convergence properties [38, 39, 69] This algorithm represents the backbone of the algorithms studied and proposed during the PhD work, which are later described in Chapter 6, and 7, with their convergence properties.

Only a few proofs are reported. For a comprehensive dissertation refer to [70, 27, 37, 38, 39, 69].

2.1 Tikhonov Regularization

2.1.1 Linear Inverse Problems

Inverse problems are pervasive in scientific and engineering fields, where the goal is to reconstruct unknown parameters or inputs from observed data. Typically, they suffer from instability, increasing the difficulty in finding a stable and accurate solution. One of the most powerful techniques to solve inverse problems has been introduced by A.N. Tikhonov, and it is well-known as Tikhonov regularization [71, 72, 73]. In this section, we discuss in a more general context Tikhonov regularization for linear inverse problems of the form:

$$Ku = g^* \tag{2.1}$$

where $K : X \to Y$ is a bounded linear operator between Banach spaces X and Y. In practice, the only available data, g^δ, are affected by noise. The accuracy to the exact data, $g^* = Ku^*$ (u^* represents the true solution), is quantified in some error metric, $\Phi(u, g^\delta)$, that measures the model output g^* relative to the measurement data g^δ.

Tikhonov regularization tackles the ill-posed problem (2.1) by solving the following well-posed optimization problem:

$$\min_{u \in C} \{J_\alpha(u) = \Phi(u, g^\delta) + \alpha\Psi(u)\} \tag{2.2}$$

and takes its minimizer, u_α^δ, as a solution [37]. The functional $J_\alpha(u)$, called Tikhonov functional, depends on the error metric, known as the *fidelity term* $\Phi(u, g^\delta)$, and the *regularization term*, $\Psi(u)$, which encodes prior

information about the solution, e.g., smoothness, sparsity. The positive scalar α represents the regularization parameter, and it balances the fidelity term and the regularization term. The choice of this parameter is fundamental to successfully applying Tikhonov regularization. The set $C \subset X$ is assumed to be convex and closed, and it represents the constraint coming from the desired solution, e.g. physical constraint.

An appropriate choice of the functionals Φ and Ψ depends on the specific applications, or more correctly, on the noise model and prior information, respectively [37].

In this thesis, we consider Gaussian additive noise, hence the L_2 fidelity function is used:

$$\Phi(u, g^\delta) = \|Ku - g^\delta\|^2 \tag{2.3}$$

where $\|\cdot\|$ represents the L_2-norm, and the noise level is such that $\delta \geq \|g^\delta - g^*\|^2$.

Well-posedness of Tikhonov Regularization

This section discusses the criteria for well-posedness, including existence, uniqueness, and stability of solutions. The most important results are stated without proof. For detailed proofs and a more comprehensive exploration of these concepts, the reader is referred to [74, 37, 27].

To proceed with the analysis of Tikhonov regularization, we state the following foundational Assumption 2.1 that ensures the well-posedness of the problem. It imposes specific conditions on the involved functionals and the properties of the underlying spaces, drawing from established results in the literature (e.g., Assumption 3.1 in [37]).

Assumption 2.1 *Let us assume the set $C \subset X$ is convex and closed, and X is reflexive. Then, the non-negative functionals $\Phi(u, g^\delta)$ (2.3), and $\Psi(u)$ satisfy the following conditions:*

(a) The functional $J_\alpha(u)$ as in (2.2) is coercive, i.e., for any sequence $\{u_n\}$ with $\{J_\alpha(u_n)\}$ uniformly bounded, the sequence $\{u_n\}$ is uniformly bounded in X.

(b) The functional $\Psi(u)$ as in (2.2) is sequentially weakly lower semicontinuous.

(c) The operator $K : X \to Y$ is bounded.

Now, let us recall the *H-property* of a functional Ψ (from Definition 3.1 in [37]).

Definition 2.1 *The functional $\Psi(u)$ satisfies the* H-property *on the space X if any sequence $\{u_n\} \subset X$ that satisfies $u_n \to u$ weakly for some $u \in X$, and $\Psi(u_n) \to \Psi(u)$ imply that u_n converges to u in X.*

Remark 2.1 *Norms on Hilbert spaces and $L^p(\Omega)$ spaces, with $1 < p < \infty$ satisfy the H-property.*

We now present the following result concerning the existence of minimizers for the functional $J_\alpha(u)$. As stated in Theorem 3.1 in [37], this guarantees the existence of at least one minimizer under the specified assumptions.

Theorem 2.1 (Existence of Minimizers) *Let Assumption 2.1 hold, then $\forall \alpha > 0$ there exists at least one minimizer u_α^δ to the functional J_α defined by (2.2).*

The next result concerns the stability of minimizers under perturbations in the data. It guarantees the convergence of minimizers as the input sequence approaches the true data and provides conditions for strong convergence. As stated in Theorem 3.2 in [37], the result is as follows.

Theorem 2.2 (Stability of Minimizers) *Let Assumption 2.1 hold. Let the sequence $\{g_n\} \subset Y$ be convergent to $g^\delta \in Y$, and u_n be a minimizer to the functional J_α as in (2.2) with g_n in place of g^δ. Then the sequence $\{u_n\}$ contains a subsequence converging to a minimizer to J_α. Moreover, if the minimizer to J_α is unique, then the whole sequence converges. Lastly, if the functional Ψ satisfies the H-property, then the convergence is strong.*

Next, the behavior of the minimizer u_α^δ, as the noise level δ approaches zero, is examined. A crucial aspect of the Tikhonov regularization method is determining whether the approximate solution u_α^δ, as the noise level δ tends to zero, converges to the true solution, i.e., the Ψ-minimizing solution u^* (defined in the following Definition 2.2 as stated in Definition 3.2 in [37]). This property is commonly referred to as consistency in the literature.

Definition 2.2 (Ψ-Minimizing Solution) *An element $u^* \in X$ is called a Ψ-minimizing solution to the problem $Ku = g^*$ if it satisfies the following:*

$$Ku^* = g^* \quad \text{and} \quad \Psi(u^*) \leq \Psi(u) \quad \forall u \in \{u \in C : Ku = g^*\}.$$

where g^* are the exact observational data.

The existence of a Ψ-minimizing solution follows from Assumption 2.1.

The following Theorem 2.3 establishes the existence of at least one Ψ-minimizing solution to the inverse problem. This result ensures that under the conditions specified in Assumption 2.1, the problem admits a solution that minimizes the regularization functional. As stated in Theorem 3.3 of [37], the result is presented below.

Theorem 2.3 *Let Assumption 2.1 hold. Then there exists at least one $\Psi-$minimizing solution to (2.1).*

Finally, the following Theorem 2.4 establishes the conditions under which minimizers converge as the noisy data sequence approaches the true data. It provides criteria for weak and strong convergence based on the behavior of the regularization parameter. As stated in Theorem 3.4 of [37], the result is presented below.

Theorem 2.4 (Consistency of Minimizers) *Let the assumption 2.1 hold. Let $\{g_n^{\delta_n}\} \subset Y$ be a sequence of noisy data satisfying $\delta_n = \|g^\delta - g^*\| \to 0$. Then the sequence of minimizers $\{u_{\alpha_n}^{\delta_n}\}$ has a subsequence converging weakly to a Ψ-minimizing solution u^*, if the regularization parameter $\alpha_n \equiv \alpha(\delta_n)$ satisfies*

$$\lim_{n\to\infty} \frac{\delta_n^2}{\alpha_n} = 0 \quad \text{and} \qquad \lim_{n\to\infty} \alpha_n = 0$$

Moreover, if the Ψ-minimizing solution u^* is unique, then the whole sequence converges weakly. Lastly, if the functional Ψ satisfies the H-property, then the convergence is actually strong.

These properties collectively validate the effectiveness of Tikhonov regularization as a robust approach for addressing ill-posed inverse problems. Building on the framework established for linear problems, the following section extends these concepts to nonlinear inverse problems.

2.1.2 Nonlinear Inverse Problems

In this paragraph, we want to deal with the nonlinear case. Nonlinear inverse problems are formulated as solving the ill-posed nonlinear operator equations

$$K(u) = g^* \tag{2.4}$$

where $K : X \to Y$ is a nonlinear operator between Banach spaces X and Y, $u \in X$ is the unknown to be determined, and $g^* \in Y$ represents the exact data.

In this case, the fidelity term is

$$\Phi(u, g^\delta) = \|K(u) - g^\delta\|^2$$

The functional $J_\alpha(u)$ as in (2.2) with $\Phi(u, g^\delta)$ as above is minimized over a closed convex subset $C \subseteq X$, ensuring that the solution adheres to additional constraints (e.g., $u \geq c$ almost everywhere).

Well-posedness of Tikhonov Regularization

As with the linear case, extending the analysis of Tikhonov regularization to nonlinear inverse problems requires establishing conditions to ensure the well-posedness of the associated functional $J_\alpha(u)$. The following Assumption 2.2 aligns with established results in the literature (see Assumption 4.1 in [37]).

Assumption 2.2 *The operator $K : X \to Y$, X being reflexive, and the non-negative functional $\Psi : Y \to \mathbb{R}^+$ satisfy*

(i) The functional $J_\alpha(u)$ as in (2.2) is coercive, i.e., $J_\alpha(u_n) \to \infty$ as $\|u_n\|_X \to \infty$.

(ii) The operator $K : X \to Y$ is sequentially weakly closed, meaning that $u_n \to u^*$ weakly in X implies $K(u_n) \to K(u^*)$ weakly in Y.

(iii) The regularization functional $\Psi(u)$ as in (2.2) is proper, convex, and weakly lower semicontinuous.

The first property to establish for nonlinear inverse problems is the existence of minimizers. The following Theorem 2.5 ensures that, under the conditions specified in Assumption 2.2, a minimizer for the functional $J_\alpha(u)$ exists for any $\alpha > 0$. As stated in Theorem 4.1 of [37], the result is presented below.

Theorem 2.5 (Existence of Minimizers) *Let the assumption 2.2 hold. Then, for any $\alpha > 0$, there exists a minimizer to $J_\alpha(u)$.*

Stability is a crucial property that ensures the minimizer of $J_\alpha(u)$ varies continuously with perturbations in the data. The following Theorem 2.6 guarantees that the minimizers of the Tikhonov functional remain stable under such perturbations. As stated in Theorem 4.2 of [37], the result is presented below.

Theorem 2.6 (Stability of Minimizers) *Let the assumption 2.2 hold. Let $\{g_n\}$ be a sequence converging to $g^\delta \in Y$, and let $\{u_n\}$ be the sequence of minimizers to J_α, with g_n in place of g^δ. Then the sequence $\{u_n\}$ contains a weakly convergent subsequence, and the limit is a minimizer to the functional J_α. Moreover, if the minimizer is unique, the entire sequence converges weakly. Lastly, if Ψ satisfies the H-property, then the convergence is strong (Theorem 4.2 in [37]).*

In addition to stability with respect to data perturbations, the Tikhonov minimizer u_α^δ must also remain stable under variations in the regularization parameter α. The following Theorem 2.7 extends the stability property by addressing the dependence of minimizers on the regularization parameter. As stated in Theorem 4.3 of [37], the result is presented below.

Theorem 2.7 *Let the assumption 2.2 hold. Let $\{\alpha_n\} \subset \mathbb{R}^+$ be a sequence converging to $\alpha > 0$ in Y, and $\{u_{\alpha_n}^\delta\}$ be the sequence of minimizers to J_{α_n}. Then the sequence $\{u_{\alpha_n}^\delta\}$ contains a weakly convergent subsequence, and the limit is a minimizer to the functional J_α. Moreover, if the minimizer is unique, the entire sequence convergences weakly, and if the functional Ψ satisfies the H-property, then the convergence is strong.*

In the context of nonlinear inverse problems, the notion of a Ψ-minimizing solution is formalised in the following Definition 2.3. The description follows directly the Definition 4.1 of [37].

Definition 2.3 *An element $u^* \in X$ is called an Ψ-minimizing solution if*

$$\Psi(u^*) \leq \Psi(u) \quad , \qquad \forall u \in \{u \in C : K(u) = g^*\}$$

The following Theorem 2.8 guarantees that, under the given assumptions, at least one Ψ-minimizing solution exists for the inverse problem. This result is directly stated in Theorem 4.4 of [37].

Theorem 2.8 *Let the assumption 2.2 hold, and there exists a solution to (2.4). Then there exists at least one Ψ-minimizing solution.*

Finally, the Theorem 2.9 addresses the consistency of minimizers, i.e., the convergence of minimizers u_α^δ to a Ψ-minimizing solution as the noise level δ tends to zero. Assumption 2.2, in connection with an appropriate rule for selecting α, is sufficient to guarantee this result. Theorem 4.5 of [37] formally states this and is presented below.

Theorem 2.9 (Consistency of Minimizers) *Let the assumption 2.2 hold. Let the sequence $\{\delta_n\}$ be convergent to zero, and g^{δ_n} satisfy $\|g^{\delta_n} - g^*\| = \delta_n$. Moreover, the parameter $\alpha(\delta_n)$ is chosen such that*

$$\lim_{\delta_n \to 0} \alpha(\delta_n) = 0 \quad \text{and} \quad \lim_{\delta_n \to 0} \frac{\delta_n^2}{\alpha(\delta_n)} = 0$$

Let $\{u_{\alpha(\delta_n)}^{\delta_n}\}$ be a sequence of minimizers to $J_{\alpha(\delta_n)}$ with g^{δ_n} in place of g^δ. Then it contains a subsequence converging weakly to a Ψ-minimizing solution. In addition, if the Ψ-minimizing solution u^* is unique, then the entire sequence converges weakly, and, finally, if the functional Ψ satisfies the H-property, then the convergence is strong.

With the theoretical framework of Tikhonov regularization for both linear and nonlinear inverse problems established, we now continue by addressing the optimization challenges associated with solving the resulting regularized inverse problems of this thesis. The Block Nonlinear Gauss-Seidel method offers an effective iterative approach by decomposing complex problems into subproblems of small size and exploiting their structure. This method provides a robust framework for constrained optimization and serves as a foundational tool in the numerical strategies developed in this thesis.

2.2 Block Nonlinear Gauss-Seidel Method

One of the most well-known sequential decomposition schemes, which can be interpreted as an extension of the Gauss-Seidel (GS) method for solving systems of linear equations, is the block nonlinear Gauss-Seidel, consisting of a sequence of global minimizations with respect to individual blocks.

Consider the problem of minimizing a function $f : \mathbb{R}^n \to \mathbb{R}$ continuously differentiable

$$\min_{x \in X} f(x) \tag{2.5}$$

where X is given by the Cartesian product of closed, nonempty, and convex subsets $X_i \subseteq \mathbb{R}^{n_i}$ for $i = 1, \ldots, m$ (with $\sum_{i=1}^m n_i = n$). Hence, $X = X_1 \times \cdots \times X_m \subset \mathbb{R}^n$. If the vector $x \in \mathbb{R}^n$ is partitioned into m component vectors $x_i \in \mathbb{R}^{n_i}$, then the problem (2.5) can be addressed through the *block non linear Gauss-Seidel* (GS) method, defined by the following iteration

$$x_i^{(k+1)} = \arg\min_{y_i \in X_i} f(x_1^{(k+1)}, \ldots, x_{i-1}^{(k+1)}, y_i, x_{i+1}^{(k)}, \ldots, x_m^{(k)}) \quad , \qquad i = 1, \ldots, m$$

that updates in turn the components of x starting from a given initial point $x^{(0)} \in X$ and generates the sequence $\{x^{(k)}\} = \{(x_1^{(k)}, \ldots, x_m^{(k)})\}$.

The GS method may not converge, producing a sequence where the limit points do not represent the critical points of the problem [75, 76].

Notation and terminology follow standard conventions.

In correspondence to the partition of the vector x, i.e., $(x_1, x_2, \ldots, x_m)$, the function value $f(x)$ is indicated by $f(x_1, x_2, \ldots, x_m)$, and the partial gradient of f with respect to x_i, evaluated at x for $i = 1, 2, \ldots, m$ is defined by $\nabla_i f(x) = \nabla_i f(x_1, x_2, \ldots, x_m) \in R^{n_i}$.

$\bar{x} \in X$ is a critical point for the problem (2.5) if $\nabla f(\bar{x})^T(y-\bar{x}) \geq 0$, $\forall y \in X$, where $\nabla f(x) \in R^n$ denotes the gradient of f at x. Considering both $\bar{x}$ and y partitioned into component vectors, $\bar{x}$ is a critical point for the problem 2.5 if and only, for all $i = 1, \ldots, m$ the following assumption is satisfied:

$$\nabla_i f(\bar{x})^T(y_i - \bar{x}_i) \geq 0 \quad , \qquad \forall y_i \in X_i$$

The level set of f relative to X corresponding to a given point $x^{(0)} \in X$ is denoted as $\mathscr{L}_X^0 : \{x \in X : f(x) \leq f(x^{(0)})\}$.

2.2.1 Feasible Descent via Line Search

To address the convergence properties of the GS method, it is essential to recall some well-known properties of an Armijo-type line search algorithm along a feasible direction, which will be utilized in subsequent analyses.

Let $\{z^{(k)}\} \in X$ be a sequence partitioned into components $z_i^{(k)} = (z_1^{(k)}, \ldots, z_m^{(k)})$, with $z_i^{(k)} \in X_i$ for $i = 1, \ldots, m$. At each iteration, the following search direction is computed

$$d_i^{(k)} = w_i^{(k)} - z_i^{(k)} \quad , \qquad w_i^{(k)} \in X_i \tag{2.6}$$

such that the following Assumption 2.3 holds.

Assumption 2.3 *Let $\left\{d_i^{(k)}\right\}$ be the sequence of search directions defined by (2.6), then:*

(i) there exists a number $M > 0$ such that $\|d_i^{(k)}\| \leq M$ $\forall k$.

(ii) $\nabla_i f\left(z^{(k)}\right)^T d_i^{(k)} < 0$ for all k.

An Armijo-type line search algorithm can be represented as follows

Algorithm 1 Armijo Line Search

1: Data: $\gamma_i \in (0,1)$, $\delta_i \in (0,1)$.
2: Compute

$$\alpha_i^{(k)} = \max_{j=0,1,\ldots} \left\{(\delta_i)^j : f(z_1^{(k)}, \ldots, z_i^{(k)} + (\delta_i)^j d_i^{(k)}, \ldots, z_m^{(k)}) \leq f(z^{(k)}) + \gamma_i(\delta_i)^j \nabla_i f(z^{(k)})^T d_i^{(k)}\right\} \tag{2.7}$$

Here, γ_i and δ_i regulate the step size.

The following Proposition 2.1 describes the behavior of the Gauss-Seidel method when combined with a line search procedure. It guarantees the existence of a finite step size that meets the acceptability condition (2.7) and characterizes the convergence of the gradient along descent directions as the sequence progresses.

Proposition 2.1 *Let $\left\{z^{(k)}\right\}$ be a sequence of points in X and let $\left\{d_i^{(k)}\right\}$ be a sequence of directions such that Assumption 2.3 is satisfied. Let $\alpha_i^{(k)}$ be computed using Algorithm 1. Then:*

(i) there exists a finite integer j such that $\alpha_i^{(k)} = (\delta_i)^j$ satisfies the acceptability condition (2.7).

(ii) if $\left\{z^{(k)}\right\}$ converges to $\bar{z}$ and $\lim_{k\to+\infty} f(z^{(k)}) - f(z_1^{(k)}, \ldots, z_i^{(k)} + \alpha_i^{(k)} d_i^{(k)}, \ldots, z_m^{(k)}) = 0$, then

$$\lim_{k\to\infty} \nabla_i f(z^{(k)})^T d_i^{(k)} = 0 \tag{2.8}$$

2.2.2 The m-Block Gauss-Seidel Method

Now, the $m-$block GS method is presented below following the scheme in [39].

Algorithm 2 Gauss-Seidel Method

1: Set $k = 0$ and $x^{(0)} = (x_1^{(0)}, \ldots, x_m^{(0)}) \in X$.
2: **repeat**
3: $k = k + 1$
4: **For** $i = 1, \ldots, m$ compute

$$x_i^{(k+1)} = \arg\min_{y_i \in X_i} f(x_1^{(k+1)}, \ldots, y_i, \ldots, x_m^{(k)}) \tag{2.9}$$

5: **End For**
6: Set $x^{(k+1)} = (x_1^{(k+1)}, \ldots, x_m^{(k+1)})$.
7: **until** `the stopping condition`.

Unless specified, we assume that the updating rule (2.9) is well-defined, and each subproblem has solutions. Moreover, let us introduce the vectors which belong to X:

$$\begin{aligned} &w(k,0) = x^{(k)} \\ &w(k,i) = (x_1^{(k+1)}, \ldots, x_{i-1}^{(k+1)}, x_i^{(k+1)}, x_{i+1}^{(k)}, \ldots, x_m^{(k)}) \quad , \qquad i = 1, \ldots, m-1 \\ &w(k,m) = x^{(k+1)} \\ &w(k,m+1) = w(k+1,1) \end{aligned}$$

By construction, for each $i \in 1, \ldots, m$, it follows from (2.9) that $w(k,i)$ is the constrained global minimizer of f in the $i-$th component subspace, hence it satisfies the following optimality condition [39]

$$\nabla_i f(w(k,i))^T (y_i - x_i^{(k+1)}) \geq 0 \quad , \qquad \forall y_i \in X_i \tag{2.10}$$

The following Propositions 2.2, and 2.3 establish fundamental properties regarding the convergence and optimality of the iterates generated by the Gauss-Seidel method. These results outline conditions ensuring the convergence of function values and describe the optimality conditions at limit points of the iterates.

Proposition 2.2 *Let us suppose that for some $i \in \{0, \ldots, m\}$ the sequence $\{w(k,i)\}$ admits a limit point $\bar{w}$, then $\forall j \in \{0, \ldots, m\}$, we have:*

$$\lim_{k \to \infty} f(w(k,j)) = f(\bar{w})$$

Proof. Let us consider an infinite subset $K \subseteq \{0, 1, \ldots, \}$, and an index $i \in \{0, \ldots, m\}$ such that the subsequence $\{w(k,i)\}_K$ converges to $\bar{w}$. Using the Algorithm 2 $f(w(k+1,i)) \leq f(w(k,i))$. Because of the continuity of the function f and the convergence of $\{w(k,i)\}_K$, then $\{f(w(k,i))\}$ has a subsequence converging to $f(\bar{w})$. Moreover, because $\{f(w(k,i))\}$ is non-increasing, it is bounded from below and it converges to $f(\bar{w})$. This follows from

$$f(w(k+1,i)) \leq f(w(k+1,j)) \leq f(w(k,i)) \quad , \qquad \text{for } 0 \leq j \leq i$$

and

$$f(w(k+2,i)) \leq f(w(k+1,j)) \leq f(w(k+1,i)) \quad , \qquad \text{for } i < j \leq m$$

□

Proposition 2.3 *If, for some $i \in \{1,\dots,m\}$, the sequence $\{w(k,i)\}$ has a limit point $\bar{w}$, then*

$$\nabla_i f(\bar{w})^T (y_i - \bar{w}_i) \geq 0 \quad , \qquad \forall y_i \in X_i \tag{2.11}$$

and

$$\nabla_{i^*} f(\bar{w})^T (y_{i^*} - \bar{w}_{i^*}) \geq 0 \quad , \qquad \forall y_{i^*} \in X_{i^*} \tag{2.12}$$

where $i^* = i(\mod m) + 1$.

Proof. Let $\{w(k,i)\}_K$ be the subsequence converging to $\bar{w}$. From (2.10), considering the continuity assumption on $\nabla_i f$, the (2.11) is obtained.

To prove the second statement (2.12), let us consider $i \in \{1,\dots,m\}$ such that $i^* = i+1$, and by contradiction, let us assume that there exists a vector $\tilde{y} \in X_{i+1}$ such that

$$\nabla_{i+1} f(\bar{w})^T (\tilde{y}_{i+1} - \bar{w}_{i+1}) < 0 \tag{2.13}$$

Then, letting

$$d_{i+1}^{(k)} = \tilde{y}_{i+1} - w(k,i)_{i+1} = \tilde{y}_{i+1} - x_{i+1}^{(k)}$$

as $\{w(k,i)\}_K$ is convergent, we obtain that the sequence $\{d_{i+1}^{(k)}\}_K$ is bounded. From (2.13) and considering the continuity assumption on $\nabla_i f$, there exists a subset $K_1 \subseteq K$ such that $\nabla_{i+1} f(w(k,i))^T d_{i+1}^{(k)} < 0$, $\forall k \in K_1$. Thus, the two sequences $\{w(k,i)\}_{K_1}$, and $\{d_{i+1}^{(k)}\}_{K_1}$ are such that Assumption 2.3 holds, provided that $\{z^{(k)}\}$ is identified as $\{w(k,i)\}_{k_1}$.

Now, $\forall k \in K_1$ let us compute $\alpha_{i+1}^{(k)}$ by Algorithm 1, then

$$f(x_1^{(k+1)},\dots,x_i^{(k+1)}, x_{i+1}^{(k)} + \alpha_{i+1}^{(k)} d_{i+1}^{(k)},\dots,x_m) \leq f(w(k,i))$$

Moreover, because $x_{i+1}^{(k)} \in X_{i+1}$, $x_{i+1}^{(k)} + d_{i+1}^{(k)} \in X_{i+1}$, $\alpha_{i+1}^{(k)} \in (0,1]$, and X_{i+1} is convex, then $x_{i+1}^{(k)} + \alpha_{i+1}^{(k)} d_{i+1}^{(k)} \in X_{i+1}$. Hence, recalling

$$f(w(k,i+1)) = \min_{y_{i+1} \in X_{i+1}} f(x_1^{(k+1)},\dots,x_i^{(k+1)}, y_{i+1},\dots,x_m^{(k)})$$

it is possible to write the following

$$f(w(k,i+1)) \leq f(x_1^{(k+1)},\dots,x_i^{(k+1)}, x_{i+1}^{(k)} + \alpha_{i+1}^{(k)} d_{i+1}^{(k)},\dots,x_m^{(k)}) \leq f(w(k,i)) \tag{2.14}$$

From Proposition 2.2, the sequences $\{f(w(k,j))\}$ are convergent to a unique limit $\forall j \in \{0,\dots,m\}$, hence

$$\lim_{k\to\infty,\, k\in K_1} f(w(k,i)) - f(x_1^{(k+1)},\dots,x_i^{(k+1)}, x_{i+1}^{(k)} + \alpha_{i+1}^{(k)} d_{i+1}^{(k)},\dots,x_m^{(k)}) = 0$$

From Proposition 2.1, having $\{z^{(k)}\}$ as $\{w(k,i)\}_{K_1}$, it follows $\nabla_{i+1} f(\bar{w})^T (\tilde{y}_{i+1} - \bar{w}_{i+1}) = 0$, that contradicts (2.13). So it follows that (2.12) has been proved when $i \in \{1,\dots,m-1\}$. When $i = m$, so that $i^* = 1$, the procedure can be repeated noting that $w(k,m+1) = w(k+1,1)$. □

The previous proposition indicates that every limit point of the sequence $\{x^{(k)}\}$ produced by the GS method is a critical point concerning the components x_1, and x_m within the established ordering. This can be formally expressed by the following corollary [39].

Corollary 2.1 *Let $\{x^{(k)}\}$ be the sequence generated by the GS method, and suppose a limit point $\bar{x}$ exists. Then,*

$$\nabla_1 f(\bar{x})^T (y_1 - \bar{x}_1) \geq 0 \quad , \qquad \forall y_1 \in X_1$$

and

$$\nabla_m f(\bar{x})^T (y_m - \bar{x}_m) \geq 0 \quad , \qquad \forall y_m \in X_m$$

Finally, in the last part of this section, we present the two-block nonlinear Gauss-Seidel method, which has been selected for analysis due to its relevance and application in the current numerical method proposed in this PhD research.

2.2.3 The Two-Block Gauss-Seidel Method

After examining the general framework of the m-block nonlinear Gauss-Seidel method, we now show the specific case where $m = 2$, referred to as the two-block Gauss-Seidel method. This case holds particular significance as it aligns with the structure of the inverse problem studied in this work, detailed later in Chapters 6 and 7. The algorithm for this method is presented below employing the notation and the form used in the proposed algorithms of the chapters of part II of this thesis.

Algorithm 3 Two-Block Gauss-Seidel Method

1: Set $k = 0$ and $x^{(0)} = (x_1^{(0)}, x_2^{(0)}) \in X$.
2: **repeat**
3: $k = k + 1$
4: Compute

$$x_1^{(k)} = \underset{\xi \in X_1}{\arg\min} f(\xi, x_2^{(k-1)}) \tag{2.15}$$

5: Compute

$$x_2^{(k)} = \underset{\xi \in X_2}{\arg\min} f(x_1^{(k)}, \xi) \tag{2.16}$$

6: **until** `the stopping condition`.
7: **return** $(x_1^{(k)}, x_2^{(k)})$

This chapter has detailed the theoretical principles underlying regularization and optimization methods for inverse problems, concluding with the presentation of the two-block Gauss-Seidel method. By iteratively updating variable subsets, this method provides a structured framework for solving constrained optimization problems, ensuring feasibility and convergence under appropriate conditions. Now, it is essential to identify a suitable strategy for computing the regularization parameter to balance model complexity and data fidelity, ultimately enhancing the stability and accuracy of the solutions. The next chapter presents strategies for the automatic computation of the regularization parameter, with particular emphasis on the Balancing and Uniform Penalty principles.

Chapter 3

Automatic Computation of the Regularization Parameter

In the previous Chapter 2, the theoretical foundations and well-posedness results have been discussed in the general context of Tikhonov regularization for linear and nonlinear problems. These results illustrate the critical role of regularization in stabilizing inverse problems. The analysis emphasizes the balance between fidelity to observed data and the incorporation of prior information, achieved via parametrized regularization terms.

To effectively determine an appropriate regularization parameter in inverse problems, various parameter selection strategies have been developed in the literature, primarily for L_2-regularization methods. Significant examples include the Discrepancy Principle, the Hanke-Raus rule, and the Quasi-optimality criterion, which have been extensively discussed in works such as [37, 27, 77, 74]. These techniques focus on balancing data fidelity and regularization to stabilize solutions. However, for example, the case of parameter selection for L_1-regularization remains less explored. Some efforts, such as [78, 79], have analyzed the applicability of the discrepancy principle in this context. Despite these advances, practical implementation remains challenging due to the need for prior knowledge of the noise level and the non-guaranteed existence of a solution to the discrepancy equation. As a consequence of these problems, in this chapter, we discuss another automatic selection rule employed in the numerical algorithms proposed in this thesis (see Chapter 5, 6, and 7): the Balancing Principle (BP). The BP determines the regularization parameter by balancing the contributions of the data fidelity and regularization terms, up to a multiplicative factor γ. This method, proposed in [80], offers a practical framework that does not require prior knowledge of the noise level and avoids solving a potentially ill-posed discrepancy equation.

Specifically, the first part of this chapter discusses the augmented Tikhonov framework extending the previous ideas through hierarchical Bayesian modeling and introducing the balancing principle to automatically search and optimize the regularization parameter and solution simultaneously.

In the final section of this chapter, a regularization framework known as the Uniform Penalty principle [40] is introduced and analyzed, focusing on a principle designed to distribute the influence of regularization terms uniformly across the domain. Unlike traditional methods that rely on a single global parameter, this approach employs a set of distinct parameters, each associated with specific points in the distribution. By enabling localized control over the regularization process, the framework effectively adapts to variations within the solution, ensuring a balanced and consistent contribution throughout. The principle has been used to derive new strategies to analyze NMRD profiles described in Chapter 5.

3.1 The Balancing Principle

This section examines the augmented Tikhonov regularization approach, emphasizing the balancing principle [80] and a convergent fixed-point iterative scheme for its realization.

The balancing principle determines the optimal regularization parameter by ensuring that the contributions of fidelity and penalty terms are harmonized. It prevents overfitting by stabilizing solutions under noisy data while preserving key features of the model.

3.1.1 Augmented Tikhonov Regularization

The augmented Tikhonov regularization extends the classical Tikhonov functional by incorporating probabilistic modelling [81]. Specifically, it minimizes the augmented functional for the maximum a posteriori estimator:

$$J(\boldsymbol{u}, \lambda, \tau) = \tau\Phi(\boldsymbol{u}, \boldsymbol{g}^\delta) + \lambda\Psi(\boldsymbol{u}) + b_0\lambda - a_0' \ln\lambda + b_1\tau - a_1' \ln\tau \tag{3.1}$$

where:

- $\Phi(\boldsymbol{u}, \boldsymbol{g}^\delta) = \frac{1}{2}\|\boldsymbol{K}\boldsymbol{u} - \boldsymbol{g}^\delta\|^2$ represents the fidelity term, penalizing the discrepancy between $\boldsymbol{K}\boldsymbol{u}$ and $\boldsymbol{g}^\delta$.
- $\Psi(\boldsymbol{u}) = \frac{1}{2}\|\boldsymbol{L}\boldsymbol{u}\|^2$ is the regularization term, enforcing specific properties via the regularization matrix $\boldsymbol{L} \in \mathbb{R}^{m' \times m}$, which is of full column rank.
- a_0, b_0, a_1, b_1 are hyperparameters derived from prior distributions of λ and τ. Specifically, $a_0' = \frac{m}{2} - 1 + a_0$, $a_1' = \frac{n}{2} - 1 + a_1$. The meaning of λ and τ is described in the next paragraph.

Considering the terms of (3.1), the first follows from an independent identically distributed Gaussian assumption on the noise in the data $\boldsymbol{g}^\delta$. The parameter τ is the inverse of the variance of the Gaussian noise, i.e., the precision. The second term assumes a Markov random field on the unknown, with the interaction structure encoded by $\boldsymbol{L}$. The parameter λ weights the strength of the interactions between neighbouring sites. The last two terms assume a Gamma distribution on the precision τ, and scale λ, with parameter pair being (a_0, b_0), and (a_1, b_1) respectively ($\lambda \sim G(\lambda; a_0, b_0)$, and $\tau \sim G(\tau; a_1, b_1)$).

This functional (3.1) is called augmented Tikhonov regularization. The first two terms reproduce the classical Tikhonov regularization described in the previous sections (with the regularization parameter $\alpha = \lambda\tau^{-1}$). The other terms give the mechanism to automatically determine the noise precision τ and the parameter λ.

The augmented formulation solves for u, λ, and τ simultaneously. The critical novelty lies in automatically determining λ and τ through Bayesian-inspired terms, reducing the need for external parameter selection.

By considering the limit of the discrete functional, the general augmented Tikhonov functional is given by

$$J(u, \lambda, \tau) = \tau\Phi(u, g^\delta) + \lambda\Psi(u) + b_0\lambda - a_0 \ln\lambda + b_1\tau - a_1 \ln\tau \tag{3.2}$$

The parameter pairs (a_0, b_0), and (a_1, b_1) should be the limit of the discrete values.

The following Definition 3.1 introduces the concept of a critical point for the functional $J(u, \lambda, \tau)$. Following Definition 3.5 in [37], the formal description is given below.

Definition 3.1 (Critical Point) *The critical point $(u^*, \lambda^*, \tau^*) \in X \times \mathbb{R}^+ \times \mathbb{R}^+$ for the functional $J(u, \lambda, \tau)$ satisfies:*

$$u^* = \arg\min_{u \in X} \left\{ \Phi(u, g^\delta) + \lambda^* (\tau^*)^{-1} \Psi(u) \right\}$$

$$\Psi(u^*) + b_0 - \frac{a_0}{\lambda^*} = 0 \quad , \qquad \Phi(u^*, g^\delta) + b_1 - \frac{a_1}{\tau^*} = 0.$$

The regularization parameter $\alpha = \lambda/\tau$ emerges naturally from the above conditions. Substituting λ^* and τ^*, one obtains:

$$\alpha^* = \frac{1}{\gamma}\frac{\Phi(u^*, g^\delta) + b_1}{\Psi(u^*) + b_0}$$

where $\gamma = \frac{a_1}{a_0}$ is determined by prior knowledge.

3.1.2 Balancing Principle

The balancing principle [80] provides a systematic approach to determine α^*. It balances the contributions of the fidelity and penalty terms by finding $\alpha^* > 0$ such that:

$$\Phi(u_{\alpha^*}, g^\delta) = \gamma\alpha^*\Psi(u_{\alpha^*}) \tag{3.3}$$

where $u_{\alpha^*} = \arg\min_{u\in X}\{\Phi(u, g^\delta) + \alpha^*\Psi(u)\}$.

Therefore, the BP selects the regularization parameter by balancing, up to a multiplicative factor γ, the contributions of the data fidelity and regularization terms. For the estimation of γ various strategies have been proposed [82, 83, 84].

It is worth highlighting the case where $\gamma = 1$, commonly referred to as the zero-crossing method, which is widely used in medical engineering applications [85, 86]. In the algorithms developed to address the parameter estimation problem within the model-free approach, as presented in Chapters 5, 6 and 7, this formulation has been consistently adopted with $\gamma = 1$.

3.1.3 Fixed-point Algorithm for Computing the Regularization Parameter

The balancing principle can be implemented via an iterative fixed-point algorithm that alternates between updating the solution u_α and the regularization parameter α. The fixed-point algorithm proceeds as follows.

Algorithm 4 Fixed Point Algorithm for Regularization Parameter

1: Set $k = 0$, and choose a starting guess α_0.
2: **repeat**
3: Set $k = k + 1$
4: Solve for $u^\delta_{\alpha_k}$ with $\alpha = \alpha_k$ by minimizing

$$u_{\alpha_k} \in \arg\min_{u\in X}\left\{\Phi(u, g^\delta) + \alpha_k\Psi(u)\right\}$$

5: Update the regularization parameter α_{k+1} by

$$\alpha_{k+1} = \frac{1}{\gamma}\frac{\Phi(u^\delta_{\alpha_k}, g^\delta)}{\Psi(u^\delta_{\alpha_k})}$$

6: **until** `the stopping condition` (3.4) is satisfied.
7: **return** approximation $(u^\delta_{\alpha_k}, \alpha_k)$.

The algorithm stops when the following condition (3.4) is satisfied.

$$|\alpha_{k+1} - \alpha_k| < \text{Tol} \quad , \qquad \text{with Tol} > 0. \tag{3.4}$$

This algorithm generates a sequence $\{\alpha_k\}$ which converges to the local minimizer as established in [80, 37].

In summary, the balancing principle provides a robust framework for determining the regularization parameter by harmonizing fidelity and penalty terms. This principle not only generalizes classical regularization

strategies but also extends their applicability to more complex models, highlighting their significance in inverse problem-solving. Starting from this principle the numerical methods and the iterative solutions studied and proposed in the following Chapters 5, 6, and 7 to analyse the NMRD profiles have been developed.

The next section introduces the Uniform Penalty Principle, which employs localized regularization parameters to adapt to variations within the solution, offering an alternative approach for addressing ill-posed inverse problems, particularly in the context of NMR data inversion.

3.2 The Uniform Penalty Principle

This last section of this chapter describes the Uniform Penalty approach [40] based on L_2 regularization with locally adapted regularization parameters.

The problem in this section is described as applied to a general NMR relaxation problem, where, considering an NMR signal generated following the formulation introduced in section 1.1.3, the purpose is to extract the relaxation time distributions for T_1, and T_2.

During the PhD work, this method has been studied and extended to the FFC-NMR estimation parameters problem and is formally presented and described in Chapter 5.

3.2.1 NMR Data Inversion Problem

NMR data are represented as a signal recorded at specific sampling points typically corresponding to evolution times. However, they can also correspond to other experiment variables, e.g., frequencies for an FFC experiment.

Let us consider a 2D NMR relaxation signal acquired through standard Inversion-Recovery, and CPMG pulse train sequence (IR-CPMG) [87]. This sequence is used to acquire 2D data by measuring both spin-lattice (T_1) and spin-spin (T_2) relaxation times. It begins with an inversion recovery (IR) pulse (generally a 180° pulse), which inverts the magnetization of the sample and allows it to recover towards equilibrium, providing insights into T_1. Following this, a series of CPMG refocusing pulses generate echoes that measure T_2. In this setup, the two evolution times, t_1 corresponding to the relaxation generated by IR, and t_2 corresponding to CPMG, are two independent variables. The measured relaxation signal can be represented as follows:

$$S(t_1, t_2) = \int \int_0^\infty k_1(t_1, T_1) k_2(t_2, T_2) F(T_1, T_2) dT_1 dT_2 + e(t_1, t_2)$$

This is a first-kind Fredholm integral equation, thus representing an ill-posed problem. Its kernel is defined by the product of the functions $k_1(t_1, T_1) = 1 - 2e^{-\frac{t_1}{T_1}}$, and $k_2(t_2, T_2) = e^{-\frac{t_2}{T_2}}$. The last term, $e(t_1, t_2)$ is the additive noise, generally modelled by a Gaussian distribution. While, the function $F(T_1, T_2)$ represents the relaxation times distribution and $F(T_1, T_2) \geq \rho$, where $\rho \in \mathbb{R}$. For this description, let us consider $\rho = 0$.

To discretize the problem, we consider $M_1 \times M_2$ sampling points for the evolution times t_1 and t_2. The resulting observations, denoted by $\boldsymbol{S} \in \mathbb{R}^{M_1 \times M_2}$, are reshaped into a vector $\boldsymbol{s} \in \mathbb{R}^M$, where $M = M_1 \times M_2$. Similarly, the unknown relaxation time distribution, initially expressed as $\boldsymbol{F} \in \mathbb{R}^{N_x \times N_y}$, is vectorized into $\boldsymbol{f} \in \mathbb{R}^N$, with $N = N_x \times N_y$. The discretized problem can now be expressed in matrix-vector form as

$$\boldsymbol{K}\boldsymbol{f} + \boldsymbol{e} = \boldsymbol{s} \tag{3.5}$$

where the matrix $\boldsymbol{K}$ is the Kronecker product of the matrices $\boldsymbol{K}_1 \in \mathbb{R}^{M_1 \times N_x}$ and $\boldsymbol{K}_2 \in \mathbb{R}^{M_2 \times N_y}$ obtained by discretization of the functions k_1 and k_2 at $M_1 \times N_x$ and $M_2 \times N_y$ points, respectively as follows

$$\boldsymbol{K} = \boldsymbol{K}_2 \otimes \boldsymbol{K}_1 \tag{3.6}$$

Finally, the vector $\boldsymbol{e} \in \mathbb{R}^M$ represents the discretization of the noise function $e(t_1, t_2)$.

3.2.2 Local Regularization through the Uniform Penalty Principle

The linear system (3.5) is an ill-conditioned inverse problem, meaning that small perturbations in the input data (e.g., noise) can lead to significant errors in the estimated solution.

To address this problem, the Tikhonov regularization (Section 2.1) is employed, which reframes it as the following minimization task

$$\min_{\boldsymbol{f}} \left\{ \|\boldsymbol{K}\boldsymbol{f} - \boldsymbol{s}\|^2 + \alpha \|\boldsymbol{L}\boldsymbol{f}\| \right\} \tag{3.7}$$

where the first term is the fidelity term, Φ, of Chapter 2. The $\|\cdot\|$ represents the L_2 norm. The second term represents the regularization term introduced in the previous chapter, Ψ. Specifically, $\boldsymbol{L} \in \mathbb{R}^{N\times N}$ is the discrete Laplacian operator, and $\alpha > 0$ is the regularization parameter which balances the data fidelity and the solution smoothness. There is no universal rule to select the best value of α. For example, let us assume to have suitable bounds on the fidelity and regularization terms of the exact solution $\boldsymbol{f}^*$, $\|\boldsymbol{K}\boldsymbol{f}^* - \boldsymbol{s}\|^2 = \varepsilon^2$, $\|\boldsymbol{L}\boldsymbol{f}\|^2 = E^2$. Then, from Miller [88], $\alpha = \frac{\varepsilon^2}{E^2}$. Using this value of the regularization parameter, the solution $\boldsymbol{f}_\alpha$ of (3.7) satisfies $\|\boldsymbol{K}\boldsymbol{f}_\alpha - \boldsymbol{s}\|^2 \leq \varepsilon^2$, and $\|\boldsymbol{L}\boldsymbol{f}_\alpha\|^2 \leq E^2$. Thus, at the regularized solution $\boldsymbol{f}_\alpha$, $\|\boldsymbol{K}\boldsymbol{f}_\alpha - \boldsymbol{s}\|^2 + \alpha\|\boldsymbol{L}\boldsymbol{f}_\alpha\|^2 \leq 2\varepsilon^2$. Having an α value such that the fidelity term and the regularization one are comparable, the result is stable in the presence of noise. While effective, Tikhonov regularization often introduces biases, particularly in regions where the solution exhibits rapid variations, giving distorted solutions which present undesired peaks.

One strategy to avoid this problem is represented by the *multiple-parameter Tikhonov regularization*, having the following minimum problem:

$$\min_{\boldsymbol{f}} \left\{ \|\boldsymbol{K}\boldsymbol{f} - \boldsymbol{s}\|^2 + \sum_{i=1}^{N} \lambda_i \left(\boldsymbol{L}\boldsymbol{f}\right)_i^2 \right\} \tag{3.8}$$

where $\left(\boldsymbol{L}\boldsymbol{f}\right)_i$ represents the $i-$th element of the vector $\boldsymbol{L}\boldsymbol{f}$. Instead of employing a single global regularization parameter α, the framework introduces a set of N distinct regularization parameters λ_i, each corresponding to a specific point within the distribution f. This approach allows for localized control over the regularization process, enabling the method to adapt to variations in the solution. The UPEN methodology establishes the values of these parameters λ_i based on the Uniform Penalty Principle, which ensures that the contributions of the regularization terms are uniformly distributed across the solution domain. This localized adjustment mitigates over-smoothing in regions of rapid variation while maintaining stability in smoother areas.

Definition 3.2 (Uniform Penalty Principle) *Choose the regularization parameters of the multiple-parameter Tikhonov regularization (3.8) such that, at a solution $\boldsymbol{f}$, the terms $\lambda_i(\boldsymbol{L}\boldsymbol{f})_i^2$ are constant $\forall i = 1, \ldots, N$ with $(\boldsymbol{L}\boldsymbol{f})_i^2 \neq 0$, i.e.*

$$\lambda_i(\boldsymbol{L}\boldsymbol{f})_i^2 = c \quad , \qquad \forall i = 1, \ldots, N \quad s.t. \qquad (\boldsymbol{L}\boldsymbol{f})_i^2 \neq 0 \tag{3.9}$$

with c positive constant.

If the non-null terms $\lambda_i(\boldsymbol{L}\boldsymbol{f})_i^2$ have all the same constant value, the regularization parameter λ_i is inversely proportional to $(\boldsymbol{L}\boldsymbol{f})_i^2$. This means that the value λ_i is smaller when $\boldsymbol{f}$ has fast changes and oscillations, while it is larger in smooth and flat regions of $\boldsymbol{f}$. Thus, regularization is enforced in points where the distribution is smooth.

The basic properties of the UPEN principle as a parameter selection rule are stated in the following lemmas.

Lemma 3.1 *If the UPEN principle holds with*

$$c = \frac{\varepsilon^2}{N_0} \tag{3.10}$$

where N_0 is the number of non-null terms $(\boldsymbol{Lf})_i^2$, and if $\boldsymbol{f}$ satisfies $\|\boldsymbol{Kf}-\boldsymbol{s}\|^2 \leq \varepsilon^2$, then

$$\|\boldsymbol{Kf}-\boldsymbol{s}\|^2 + \sum_{i=1}^{N} \lambda_i (\boldsymbol{Lf})_i^2 \leq 2\varepsilon^2 \tag{3.11}$$

Contrarily, any $\boldsymbol{f}$ which satisfies (3.11), and the UPEN principle with (3.10), also satisfies $\|\boldsymbol{Kf}-\boldsymbol{s}\|^2 \leq \varepsilon^2$.

Proof. Let $\boldsymbol{f}$ be such $\|\boldsymbol{Kf}-\boldsymbol{s}\|^2 \leq \varepsilon^2$, then if the UPEN principle is satisfied with (3.10), it holds the following

$$\|\boldsymbol{Kf}-\boldsymbol{s}\|^2 + \sum_{i=1}^{N} \lambda_i (\boldsymbol{Lf})_i^2 \leq \varepsilon^2 + \sum_{i=1}^{N_0} \frac{\varepsilon^2}{N_0} = 2\varepsilon^2$$

Contrarily, if the UPEN principle with (3.10), and the (3.11) hold, then

$$2\varepsilon^2 \geq \|\boldsymbol{Kf}-\boldsymbol{s}\|^2 + \sum_{i=1}^{N} \lambda_i (\boldsymbol{Lf})_i^2 = \|\boldsymbol{Kf}-\boldsymbol{s}\|^2 + \sum_{i=1}^{N_0} \frac{\varepsilon^2}{N_0} = \|\boldsymbol{Kf}-\boldsymbol{s}\|^2 + \varepsilon^2$$

□

From this, every solution of (3.8), f_λ, with every λ_i component chosen through the UPEN principle, is feasible with respect to the data-fidelity constraint $\|\boldsymbol{Kf}-\boldsymbol{s}\|^2 \leq \varepsilon^2$.

Lemma 3.2 *Let us define the operator*

$$R_\lambda = \left(\boldsymbol{K}^T\boldsymbol{K} + \boldsymbol{L}^T\boldsymbol{D}\boldsymbol{L}\right)^{-1}\boldsymbol{K}^T$$

with $\boldsymbol{D}$ the diagonal matrix with diagonal elements

$$D_{i,i} = \begin{cases} \lambda_i, & \text{if } (\boldsymbol{Lf})_i \neq 0 \\ \gamma\varepsilon^2, & \text{otherwise} \end{cases}$$

where γ is a positive constant and the λ_i are chosen according 3.10. Then

$$\lim_{\varepsilon\to 0} R_\lambda \boldsymbol{Kf} = \boldsymbol{f}$$

Proof. From 3.9, and 3.10, it follows

$$\lambda_i = \frac{\varepsilon^2}{N_0(\boldsymbol{Lf}_i)^2} \quad \forall \quad i = 1,\ldots,N \quad \text{such that} \quad (\boldsymbol{Lf}_i)^2 \neq 0$$

Thus, the proof immediately follows since for all i

$$\lim_{\epsilon\to 0} D_{i,i} = 0$$

□

This result states that f_λ is a regularized solution of the problem (3.5).

The following iterative scheme (Algorithm 5) has been proposed by Bortolotti et al. [40], consisting of, starting from an initial guess $\boldsymbol{f}^{(0)}$, both a solution to (3.8) and suitable values of λ_i, approximately satisfying the UPEN principle, are computed.

Algorithm 5 Iterative Scheme UPEN

1: Compute $\lambda_i^{(k)} = \frac{\|\boldsymbol{Kf}^{(k)}-\boldsymbol{s}\|^2}{N_0^{(k)}\left(\boldsymbol{Lf}^{(k)}\right)_i^2}$, with $N_0^{(k)}$, the number of non-null terms of $(\boldsymbol{Lf}^{(k)})_i^2$.
2: Compute $\boldsymbol{f}^{(k+1)}$ by solving (3.8) with $\lambda_i = \lambda_i^{(k)}$.
3: Set $k = k+1$.

In the Algorithm 5, the $k-$th residual norm $\|\boldsymbol{K}\boldsymbol{f}^{(k)} - \boldsymbol{s}\|$ is employed to approximate ε that, in case of noisy data, it is the noise norm $\|\boldsymbol{e}\|$.

When in the first step, one of the terms $(\boldsymbol{L}\boldsymbol{f}^{(k)})_i$ is negligible, it is not possible (or not meaningful) to make λ_i large enough to maintain a truly uniform penalty at such points. Moreover, a term $(\boldsymbol{L}\boldsymbol{f}^{(k)})_i$ could be equal to zero in non-flat regions due to noise and approximation errors generated throughout the iterations.

Consequently, to have more information about the shape of the distribution, it may be helpful to relax the strict uniform-penalty requirement by considering both second- and first-order derivative information in a neighborhood of the $i-$th point in the selection rule.

Let us define the matrix $\boldsymbol{C}$ of dimension $N_x \times N_y$, such that lexicographically reorders its element and gives the vector $\boldsymbol{L}\boldsymbol{f}$. Furthermore, let us introduce another matrix $\boldsymbol{P}$ of same dimensions, with elements $P_{\ell,\mu} = \|\nabla \boldsymbol{F}_{\ell,\mu}\|$. Finally, let us denote with $\boldsymbol{c}$, and $\boldsymbol{p}$ the N vectors obtained by reordering the elements of $\boldsymbol{C}$ and $\boldsymbol{P}$.

Therefore, the regularization parameters $\lambda_i^{(k)}$ are derived according to the following relaxed UPEN principle [40]:

$$\lambda_i^{(k)} = \frac{\|\boldsymbol{K}\boldsymbol{f}^{(k)} - \boldsymbol{s}\|^2}{N\left(\beta_0 + \beta_p \max_{\mu\in I_i}\left(\boldsymbol{p}_\mu^{(k)}\right)^2 + \beta_c \max_{\mu\in I_i}\left(\boldsymbol{c}_\mu^{(k)}\right)^2\right)} \quad , \qquad i = 1, \ldots, N \tag{3.12}$$

where the I_i are the indices subsets related to the neighborhood of the $i-$th entry. The β parameters are positive. Specifically, the β_0 parameter prevents division by zero and is a compliance floor, which should be small enough to prevent undersmoothing, and large enough to avoid oversmoothing. The optimum values of β parameters could substantially change with the nature of the measured sample. As a general rule, β_0 should be considerably smaller than the two other β parameters, which should be of the same order of magnitude.

The regularization parameters obtained by 3.12 are locally adapted: the selection of the values λ_i is based on local information about the shape of the desired solution.

This chapter has presented the theoretical methods for the automatic computation of regularization parameters, including the balancing principle and the uniform penalty principle. These approaches provide robust frameworks for addressing ill-posed inverse problems by ensuring stability, adaptability, and consistency with problem-specific constraints. The methodologies analyzed here form the foundation for the strategies proposed in this work, which are described in detail in the following chapters.

The next chapter explores the intersection of traditional optimization techniques and emerging data-driven approaches, highlighting the role of machine learning in advancing inverse problem-solving frameworks.

Chapter 4

Machine Learning for NMR

Machine learning has become a powerful tool for addressing problems in every domain, particularly when deriving explicit solutions from physical models could be challenging. Moreover, through statistical learning principles, it is possible to derive a mathematical formulation for machine learning. The first part of this chapter is focused on this argument.

Furthermore, the integrated models, which combine traditional optimization methods with data-driven techniques are presented in the second part of the chapter. Among the described strategies, a specific focus in the last part is dedicated to the Plug-and-Play (PnP) method, which represents a very powerful tool in imaging and it is based on the integration of pre-trained denoisers into iterative algorithms to decouple the data acquisition process from the learned priors. This description serves as a starting point to present in the further chapters the novel proposed method developed during this PhD work, which is based on the philosophy of the PnP method.

4.1 The Mathematical Principles of Machine Learning

This section outlines the foundational principles of machine learning, establishing a basis for the topics addressed in the subsequent chapters. Specifically, the problem of supervised learning is formalized through the framework of statistical learning theory [89, 90]. As this work focuses on a specific class of integrated models, a comprehensive review of all employed machine learning methodologies is beyond the scope of this thesis work.

4.1.1 Statistical Learning Framework

Let $\boldsymbol{x}$ be the input vector of $X \subseteq \mathbb{R}^p$ which represents the space of all possible inputs described with an unknown probability $p(\boldsymbol{x})$. Let $\boldsymbol{y}$ be the output vector, where $\boldsymbol{y} \in Y$, and $Y \subseteq \mathbb{R}^q$ is the space of the possible outputs. The output $\boldsymbol{y}$ for every input $\boldsymbol{x}$ is provided, according to an unknown, but fixed conditional probability $p(\boldsymbol{y}|\boldsymbol{x})$. The purpose of the supervised learning approach is to approximate a joint probability distribution $p(\boldsymbol{x}, \boldsymbol{y}) = p(\boldsymbol{x})p(\boldsymbol{y}|\boldsymbol{x})$, called data generating distribution, and which represents the relationship between X, and Y, to derive $\boldsymbol{y}$, starting from $\boldsymbol{x}$.

Let $\ell : Y \times Y \to \mathbb{R}$ be the loss function to compute the distance between two elements in Y, the expected risk of a function $f : X \to Y$ is defined as follows:

$$\mathcal{J}(f) = \mathbb{E}\left[\ell(\boldsymbol{y}, f(\boldsymbol{x}))\right] = \int_{X \times Y} \ell(\boldsymbol{y}, f(\boldsymbol{x})) dp(\boldsymbol{x}, \boldsymbol{y}) \tag{4.1}$$

Identifying a function that minimizes (4.1) represents the goal of the learning task and a performance selection criterion. Moreover, because generally ℓ is a distance function, the equation (4.1) can be rewritten using the conditional expectation as follows:

$$\mathcal{J}(f) = \mathbb{E}\left[\mathbb{E}\left[\ell(\boldsymbol{y}, f(\boldsymbol{x}))|\boldsymbol{x}\right]\right] = \int_X \mathbb{E}\left[\ell(\boldsymbol{y}, f_\theta(\boldsymbol{x}))|\boldsymbol{x} = \bar{\boldsymbol{x}}\right] dp(\bar{\boldsymbol{x}})$$

The expected risk is minimized at the Bayes predictor $f^* : X \to Y$, which pointwise is defined as:

$$f^*(\bar{\boldsymbol{x}}) \in \arg\min_{\boldsymbol{z}\in Y} \mathbb{E}\left[\ell(\boldsymbol{y}, \boldsymbol{z})|\boldsymbol{x} = \bar{x}\right] \quad , \qquad \forall \bar{\boldsymbol{x}} \in X \tag{4.2}$$

The Bayes risk correspondent to the Bayes predictor f^* is given by:

$$\mathcal{J}^* = \mathbb{E}\left[\inf_{\boldsymbol{z}\in Y} \mathbb{E}\left[\ell(\boldsymbol{y}, \boldsymbol{z})|\boldsymbol{x} = \bar{\boldsymbol{x}}\right]\right] \tag{4.3}$$

The Bayes predictor minimizes the expected risk, and while it is not necessarily unique, all such predictors yield the same Bayes risk. To evaluate the performance of a given function f, the deviation from this minimum, referred to as the excess risk of f, is quantified by $\mathcal{J}(f) - \mathcal{J}^*$. This quantity is inherently non-negative, from (4.3).

Theoretically, the optimal model is described in (4.2) by the conditional distribution $p(\boldsymbol{y}|\boldsymbol{x})$ for any input $\boldsymbol{x} \in X$. However, the underlying distribution $p(\boldsymbol{x}, \boldsymbol{y})$ is often unknown in practical applications, as only a finite observation set is usually available. To address this, the empirical risk $\widehat{\mathcal{J}}_{\mathcal{S}}(f)$, an estimate based on available training samples, is used. The training samples are in a set, commonly called *training set*, which can be formally expressed as

$$\mathcal{S} = \{(\boldsymbol{x}_i, \boldsymbol{y}_i) \in X \times Y | i = 1, \ldots, s\}$$

where each pair $(\boldsymbol{x}_i, \boldsymbol{y}_i)$ represents a sample drawn from the joint distribution $p(\boldsymbol{x}, \boldsymbol{y})$. Using this finite dataset $\mathcal{S}$, one can compute the empirical risk of a predictor f, which averages the loss function over the training samples. This is given by:

$$\widehat{\mathcal{J}}_{\mathcal{S}}(f) = \frac{1}{s} \sum_{i=1}^{s} \ell(\boldsymbol{y}_i, f(\boldsymbol{x}_i))$$

The empirical risk is an unbiased estimate of the true expected risk and often is used as a surrogate objective in learning tasks, given that the true distribution is typically inaccessible. Moreover, it is commonly called *training error*.

To address this, the learning process is recast as an optimization problem, where the search space is limited to a parametrized family of functions, referred to as the *hypothesis space*, or, identically on the parameter space. This is defined as:

$$\mathcal{H} := \{f_\theta : X \to Y | \theta \in \Theta\}$$

where $\Theta \in \mathbb{R}^p$ represents the space of all the possible parameter values. Restricting the learning to this family allows practical implementation while preserving the goal of minimizing risk. Therefore, the learning problem is reformulated by the minimization of the empirical risk on the hypothesis space as

$$\theta^* \in \arg\min_{\theta\in\Theta} \widehat{\mathcal{J}}_{\mathcal{S}}(f_\theta) \tag{4.4}$$

By constraining the search to parametrized functions within the hypothesis space $\mathcal{H}$, an approximation error is introduced which can be derived by decomposing the excess risk of f_θ^* as follows:

$$\mathcal{J}(f_{\theta^*}) - \mathcal{J}^* = \underbrace{\left(\mathcal{J}(f_{\theta^*}) - \inf_{\theta\in\Theta} \mathcal{J}(f_\theta)\right)}_{\text{estimation error}} + \underbrace{\left(\inf_{\theta\in\Theta} \mathcal{J}(f_\theta) - \mathcal{J}^*\right)}_{\text{approximation error}} \tag{4.5}$$

The last term is known as approximation error. It is always non-negative, not depending on the optimization parameters, but only on the chosen class of parametrized functions. Moreover, it evaluates how closely the functions f_θ can approximate a Bayes estimator. This error can be reduced as the considered Θ is large enough, e.g., in the case of the neural networks; furthermore, the functions can be arbitrarily approximated by the chosen parametric function [91].

Instead, the first term of (4.5) is known as estimation error and can be further decomposed, taking into account that $\bar{\theta} \in \arg\min_{\theta\in\Theta} \mathcal{J}(f_\theta)$ the following relation holds:

$$\mathcal{J}(f_{\theta^*}) - \mathcal{J}(f_{\bar{\theta}}) = \left(\mathcal{J}(f_{\theta^*}) - \widehat{\mathcal{J}}(f_{\theta^*})\right) + \underbrace{\left(\widehat{\mathcal{J}}_{\mathcal{S}}(f_{\theta^*}) - \widehat{\mathcal{J}}_{\mathcal{S}}(f_{\bar{\theta}})\right)}_{\text{empirical optimization error}} + \left(\widehat{\mathcal{J}}_{\mathcal{S}}(f_{\bar{\theta}}) - \mathcal{J}(f_{\bar{\theta}})\right)$$

The second term is called empirical optimization error, and it is bounded by $\sup_{\theta\in\Theta}\left(\widehat{\mathcal{J}}_{\mathcal{S}}(f^*_\theta) - \widehat{\mathcal{J}}_{\mathcal{S}}(f_\theta)\right)$, which in principle should be zero for the minimality assumption on θ^*, but in practice, it must be considered when iterative optimization algorithms are adopted to minimize the empirical risk. Finally, the other two terms are bounded by $\sup_{\theta\in\Theta}\left|\widehat{\mathcal{J}}_{\mathcal{S}}(f_\theta) - \mathcal{J}(f_\theta)\right|$ that increases as the size of Θ increases, and decreases with the samples number in $\mathcal{S}$.

4.1.2 Generalization

The process of learning is framed as an empirical risk minimization problem. This formulation, while conceptually intuitive and mathematically straightforward, faces significant challenges in practice. The primary difficulty lies in developing models that can accurately predict outcomes for new, unseen data that was not part of the training set. This crucial ability to perform well on previously unobserved data is referred to as *generalization*. While empirical risk provides a measure of performance on the training data, it may fail to capture the generalization capability of a model, which is the ultimate objective in most real-world applications.

To evaluate generalization, it is standard practice to partition the dataset into two disjoint subsets: the training set and the test set. The training set is used to learn the model, while the test set is reserved for evaluating its performance on independent samples drawn from the same underlying distribution. This separation ensures that the evaluation reflects the model's ability to generalize beyond the examples seen during training. The ultimate goal of any learning task is to minimize the *generalization error*, defined as the empirical risk computed on the test set, although during the training, only the empirical risk on the training set is minimized. Therefore, for a model to exhibit strong generalization, it must achieve two critical conditions: a low training error, and a minimization of the training-test error gap.

If these conditions are not jointly achieved, the model is likely to suffer from one of two classical issues in machine learning: *underfitting*, or *overfitting*. Underfitting occurs when the model struggles to achieve a low training error, often due to its inability to capture the underlying patterns in the data. Overfitting, on the other hand, arises when the model learns the training data too well, including its noise and idiosyncrasies, leading to a significant gap between training and test errors. In this case, the model performs poorly on unseen data, failing to generalize effectively.

To mitigate underfitting and overfitting, a crucial strategy involves controlling the *capacity* of the model. The term capacity loosely describes the power of a model to represent a diverse range of functions. Models with insufficient capacity may lack the flexibility needed to fit the training data, while those with excessive capacity risk overfitting by adhering too closely to the training set.

It is essential to distinguish between two aspects of model capacity:

1. The *representational capacity* refers to the size of the hypothesis space $\mathcal{H}$ and the model's ability to represent a wide class of functions. One practical approach to regulating representational capacity is through regularization techniques, which constrain the model's parameters either by limiting their magnitude or by introducing additional terms in the objective function. A regularized empirical risk minimization problem can be expressed as follows:

$$\theta^* \in \arg\min_{\theta\in\Theta} \left\{ \widehat{\mathcal{J}}_{\mathcal{S}}(f_\theta) + \lambda\Psi(\theta) \right\} \tag{4.6}$$

   where λ represents a regularization term previously described in Chapter 2, and $\Psi(\theta) : \Theta \to \mathbb{R}^+$ represents the regularization function implicitly controlling the complexity of f_θ.

2. The *effective capacity*. This concept pertains to the actual capacity of the model as influenced by the optimization algorithm used during training. Optimization methods, particularly iterative algorithms like gradient descent, often impose additional constraints on the function representation, which can lead to approximations of the true minimizer. In the realm of deep learning, determining the effective capacity is particularly challenging due to the non-convex nature of the loss landscape and the limited theoretical understanding of stochastic optimization methods, such as those involving momentum or adaptive learning rates.

Quantifying model capacity and its effect on generalization has been a focus of theoretical research. Various studies have sought to derive formal bounds to characterize the discrepancy between training error and generalization error [92, 93]. Typically, this discrepancy is bounded by a term that grows with the model's capacity and decreases as the size of the training dataset increases. Thus, a trade-off emerges between the expressiveness of the model and the availability of data. In the context of modern deep learning systems, where models often have millions or even billions of parameters, this balance becomes even more critical.

4.1.3 Optimization Techniques for Minimizing Empirical Risk

To minimize the empirical risk (4.6) iterative first-order methods are applied because the computation of a closed-form solution is not practicable. The most common and well-known as well as efficient algorithm used to derive the gradient of the objective function is the *backpropagation algorithm*, introduced in separate works at the same time, such as [94, 95], and later refined in [96].

Often referred to as the reverse mode of automatic differentiation, this approach gained widespread adoption in the learning framework through [97, 98]. Today, it serves as a core technique in popular neural network libraries and toolboxes like TensorFlow, and PyTorch for Python, but also Deep Learning Toolbox [99] in MATLAB. The general update rule for a first-order gradient-based method typically follows this form:

$$\theta^{(k+1)} = \theta^{(k)} - \alpha \boldsymbol{D}_k g_k(\theta^{(k)})$$

where α represents the so-called *learning rate* in the machine learning research community, $\boldsymbol{D}_k \in \mathbb{R}^{p\times p}$ is a diagonal matrix, and finally $g_k : \mathbb{R}^p \to \mathbb{R}^p$ is dependent on the empirical risk gradient.

The simplest gradient-based approach is standard gradient descent, where updates move in the direction opposite to the gradient [76]. In this case, the function $g_k(\theta^{(k)}) = \nabla_\theta \widehat{\mathcal{J}}_{\mathcal{S}}(f_{\theta^{(k)}})$, and the diagonal matrix $\boldsymbol{D}_k = \boldsymbol{I}_p$. However, when dealing with large datasets, stochastic adaptations of gradient descent are preferred due to their efficiency [100, 101]. In these methods, a randomly selected subset of the training data, referred to as *minibatch*, is used to compute an approximate gradient at each iteration. By drawing s samples from $\mathcal{S}$, the function g_k is computed at each iteration as follows:

$$g_k(\theta) = \frac{1}{s} \sum_{(\boldsymbol{x},\boldsymbol{y})\in\mathcal{S}_k} \nabla_\theta \ell\left(\boldsymbol{y}, f_\theta(\boldsymbol{x})\right)$$

where $\mathcal{S}_k \subset \mathcal{S}$ is a subset of the training set $\mathcal{S}$ consisting of s samples drawn at each iteration. This reduces the computational cost compared to evaluating the gradient over the entire dataset. However, the approximated gradient inherently includes noise, introducing some variability into the optimization trajectory.

Such stochastic approaches, while computationally efficient, come with trade-offs. The gradient estimates, derived from minibatches, do not always point directly toward minimizing the empirical risk. Instead, the randomness injected into the process can lead to significant fluctuations in the objective function, with no guarantee that the algorithm will converge to a minimum or even a stationary point.

The size of the minibatches used during training plays a critical role in gradient estimation. Larger minibatches tend to yield more precise gradient calculations, while smaller ones, though noisier, often introduce a beneficial regularization effect throughout the optimization process [102]. To mitigate the variance in these gradient estimates, advanced techniques leverage both first-order and second-order moment approximations, constructing appropriate scaling of the matrix $\boldsymbol{D}_k$ to enhance stability and convergence.

Given the inherent non-convexity of deep learning problems, momentum-based optimization methods have been developed to address avoiding entrapment in local minima. By incorporating information about previous gradient updates, these methods expedite learning and help maintain a steady progression toward more promising regions of the loss landscape [103, 104].

Among the most widely adopted optimization techniques for training deep neural networks is the Adam optimizer [105], which effectively combines inertia-like updates with mechanisms to reduce variance in the gradients. This method relies on the computation of specific scaling matrix $\boldsymbol{D}_k$, and gradient estimates g_k, significantly improving the efficiency and robustness of the training process.

Despite the empirical success of algorithms like Adam in minimizing empirical risk or finding stationary points, the *no free lunch* theorem asserts that, when averaged across all potential data-generating distributions, every optimization algorithm achieves the same error rate [106, 107]. This fundamental limitation implies that designing highly effective algorithms for specific applications often requires assumptions about the underlying data-generating process.

4.1.4 Architectural Frameworks for Neural Networks

In machine learning, the hypothesis space chosen for a model profoundly impacts its ability to represent data, as this determines the achievable approximation error (4.5). Over the past decade, artificial neural networks (ANNs) have gained prominence due to their versatility in approximating complex functions. These networks mimic biological neural systems and consist of a series of layers, each functioning as a set of units, known as neurons, interconnected to adjacent layers. Several architectures can be obtained by organizing the layers in different configurations. The first layer is called the input layer, and it receives input data; while the last one is called the output layer and it is responsible for producing the output of the network. The layers between these two are called hidden layers.

The most simple architecture consists of a fully connected layer, where each neuron in one layer is connected to every neuron of the previous layer. This relationship is represented by the following:

$$\bar{\boldsymbol{z}} = \sigma\left(\boldsymbol{W}\boldsymbol{z}\right) \tag{4.7}$$

where $\bar{\boldsymbol{z}} \in \mathbb{R}^m$, $\boldsymbol{z} \in \mathbb{R}^n$ represent the output and the input data respectively, and $\boldsymbol{W} \in \mathbb{R}^{m \times n}$ is the weight matrix defining the connection strengths between two consecutive layers. The function $\sigma : \mathbb{R}^n \to \mathbb{R}^m$ is the activation function applied elementwise.

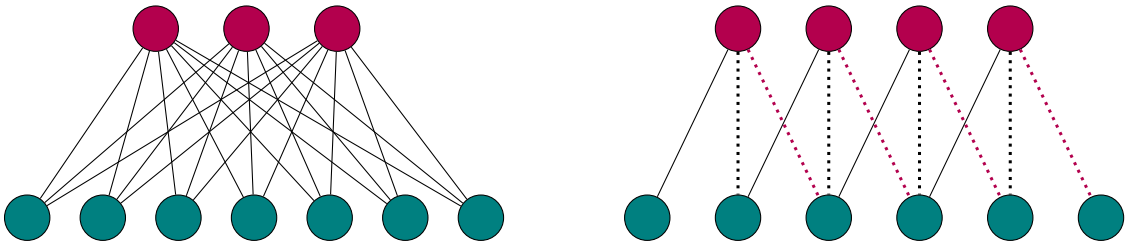
The fully connected layer has theoretical significance because of the universal approximation theorem, which asserts that even shallow networks can approximate any continuous function defined on a compact domain, provided they are equipped with sufficient neurons [108, 109]. However, achieving such approximations in practice often requires an enormous number of neurons. A practical solution to enhance representational

capacity without excessively increasing parameters is represented by the use of deeper networks formed by stacking several fully connected layers. These are called *Deep Fully Connected Network* [110, 111]. Despite their versatility, this architecture experiences exponential growth in the number of weights as the number of layers increases, resulting in the training being computationally demanding [112].

To overcome this limitation, convolutional layers were developed, which rely on localized connections instead of full connectivity. This architecture is briefly described given its relevance in this research area, but it does not represent the purpose of this thesis, because, considering the dimension of the addressed problem, a deep neural network has been sufficient, without demanding very long and complex training.

Conceptually, a convolutional layer can be thought of as a fully connected layer, wherein the weight matrix $\boldsymbol{W}$ is a sparse Toeplitz matrix [113]. This structure reduces the number of parameters significantly by sharing weights within localized regions of the input, i.e., the outputs are connected only within a local region of the input, decreasing the number of weights. These regions are commonly referred to as convolutional kernels. These kernels are particularly effective at identifying localized patterns, such as edges in images [114].

A representation of the two layers structures is shown in Figure 4.1, with the fully connected one on the left, and the convolutional one on the right.



(a) (b)

Figure 4.1: Comparison of neural network layers: (a) Fully connected layer, where each neuron is connected to all neurons in the next layer, and (b) Convolutional layer, where local connections are represented by limited links between layers. In red the neurons of the input layer, i.e., $\boldsymbol{z}$ in (4.7), while in green the neurons of the output one, i.e., $\bar{\boldsymbol{z}}$ in (4.7).

Convolutional layers also assure shift-invariance, enabling the detection of features regardless of their spatial location. This property accelerates learning while significantly reducing computational complexity [115].

Neural networks that include at least one convolutional layer are known as Convolutional Neural Networks (CNNs). The foundational concepts of CNNs were inspired by studies of the primary visual cortex [116, 117].

The performance of machine learning methods depends generally on the data representation utilized. Therefore, it is common practice to encode raw data to extract meaningful features for the learning task at hand. This process can be particularly challenging since the most important features are not known in advance. As a result, the model must learn the appropriate representations itself. The complexity of the extracted features depends on the task's difficulty, with high-level abstract features often being formed by combining simpler, lower-level blocks. The approach that utilizes multiple layers in the network is known as Deep Learning. Among its many applications, Convolutional Neural Networks (CNNs) are particularly important due to their capacity to learn hierarchical feature representations directly from complex data. Furthermore, the use of convolutional layers with smaller receptive fields enabled the development of deeper CNN architectures, increasing their capacity to learn from vast amounts of data. These deep architectures have led to an expanded receptive field, allowing models to capture long-range dependencies.

To further enhance training, additional strategies such as residual blocks and skip connections are utilized to maintain information flow and address issues like vanishing gradients [118]. Batch normalization is another

commonly employed method, which stabilizes training by normalizing mini-batch inputs [119]. The use of diverse activation functions introduces non-linearity, profoundly affecting both model performance and training efficiency [120]. Finally, techniques like weight initialization and regularization are employed to combat overfitting and improve the model's robustness [92].

4.2 Integrated Models

One of the most interesting approaches in this learning process involves using a direct inversion method. In this way, it is possible to incorporate a physical knowledge of the acquisition model [121].

When the direct inversion of the acquisition model is feasible, one can employ preprocessing techniques to map the given measurement $\mathbf{b}$, e.g., a signal such as the relaxation rate, $\boldsymbol{R}_1$, introduced in Chapter 1, into the correlation time distribution, $\boldsymbol{f}$, domain through operations such as $\boldsymbol{K}^T\boldsymbol{R}_1$ or $\boldsymbol{K}^\dagger\boldsymbol{R}_1$. These methods have been introduced and deeply investigated specifically for imaging problems, where the idea is based on consists of obtaining a naive reconstruction with artefacts and then using a trained neural network to correct the approximated restoration [122, 123, 124]. However, following this philosophy, this method can also be extended to different domains. Therefore, the last contribution presented in this PhD thesis is based on the idea behind this class of approaches and it is comprehensively described in Chapter 7.

Mathematically, the learned operator can be expressed as the composition $f_\theta \circ \boldsymbol{K}^T$, defining $f_\theta : \mathbb{R}^n \to \mathbb{R}^m$ the trained neural network to correct the initial estimate. The problem can be reformulated as follows:

$$\theta^* \in \arg\min_\theta \frac{1}{s}\sum_{i=1}^{s} \ell\left(f_\theta\left(\boldsymbol{K}^T\boldsymbol{b}_i\right), \boldsymbol{u}_i\right)$$

where ℓ is a suitable loss function, $\boldsymbol{b}_i$ are the measurements, and $\boldsymbol{u}_i$ represent the corresponding ground-truth signals. However, a common problem is represented by overfitting. The usual strategy involves the introduction of a regularization term, trying to enforce specific characteristics of the reconstruction, which may be a priori known. Following this, the problem can be rewritten as:

$$\theta^* \arg\min_\theta \frac{1}{s}\sum_{i=1}^{s} \left[\ell\left(f_\theta\left(\boldsymbol{K}^T\boldsymbol{b}_i\right), \boldsymbol{u}_i\right) + \lambda\Psi\left(f_\theta\left(\boldsymbol{K}^T\boldsymbol{b}_i\right)\right)\right]$$

where λ represents the regularization parameter, and Ψ the function inducing some bias to the reconstruction. Generally, the introduction of the regularization term is tricky because it involves strong assumptions underlying the distribution. Moreover, a good choice of the regularization parameter can only be estimated after the training process.

In recent years, numerous improvements have been made in merging neural networks and classical variational methods, paving the way for hybrid approaches. The strength of these models is represented by the capability to leverage the expressiveness of deep learning while maintaining the interpretability and robustness of traditional frameworks. This led to enhanced trustworthiness in these models and reduced dependence on training data, especially for inverse problems.

The development of integrated models remains an active area of research with an increasing number of contributions. A comprehensive review of all approaches is beyond the scope of this thesis. Therefore, in the next section different techniques and their applications in inverse problems are mentioned and briefly described, before introducing the integrated model which has been the starting point for the work developed in this PhD: the *Plug-and-Play* (PnP) method.

4.2.1 Key Techniques in Integrated Modelling

Unfold Methods This method consists of unfolding and unrolling an iterative procedure for a finite number of steps. The iteration can be thought of as a layer of a neural network. By the concatenation of

them, it is possible to build a deep neural network. This allows this framework to learn complex mappings between output and input, maintaining interpretability. Essentially, this strategy is identical to running an iterative procedure for a select number of iterations. This method is trained on a specific training set, considering the parameters as the network hyperparameters. This approach has been introduced by LeCun and Gregor [125] and during the last years has been improved showing its potential in several domains, i.e., inverse problems, image reconstruction, and also signal processing [126, 127, 128, 129, 130].

Bilevel Methods Bilevel optimization represents a class of integrated models based on the idea of solving two interdependent optimization problems at the same time. One is an upper-level problem to find the optimal parameters for the model, the other is a lower-level problem to find the optimal solutions. The first one is generally a training to minimize the loss function (4.4), and the latter one represents usually a specific optimization problem. By using this method, the best hyperparameters to minimize the loss function are refined while the model is adapting to the data distribution. A review of this method can be found in [131].

Generative Methods This class of methods is profoundly important in the entire artificial intelligence sector because able to mime and understand very complex data distributions. Examples of generative models are represented by Generative Adversarial Networks (GANs), or Variational Autoencoders (VANs), which with advanced training strategies, and architectures succeed in learning the underlying data distributions.

The idea behind the GANs is to exploit two neural networks competing against each other in a zero-sum game, where a gain for one of them represents a loss for the other one. Starting from a training set, this technique learns how to create new data having the same statistics as the initial training set. The training consists of employing a generator, i.e., one of the two neural networks, to create "realistic" input, and a discriminator, i.e., the other neural network, to evaluate their authenticity. This allows the model to learn in an unsupervised mode. This method has been formulated by Goodfellow and its formalization can be found in [132].

The VANs instead focus on learning a probabilistic latent space to generate realistic reconstruction. [133].

Building on the advantages of integrated modelling, the PnP method offers a flexible solution for inverse problems by decoupling data acquisition from learned priors.

4.2.2 Plug-and-Play

The Plug-and-Play (PnP) method has been proposed by Venkatakrishnan et al. [134] and it represents an effective strategy to solve inverse problems integrating model-based optimization with data-driven priors.

One of the main problems related to this learning method is the stability of the model [135]. Specifically, the noise in the data can drive the reconstruction to produce artifacts [136]. Moreover, in contrast to variational approaches, the learning ones need to be retrained every time the acquisition model changes.

This problem has been addressed by the PnP method, creating a flexible model which involves separating the degradation process from the learned prior. The idea behind this method is to employ the modularity of iterative algorithms, integrating available denoisers methods. This allows the method to obtain robust reconstructions without relying on specific training data, extending this technique to several different problems.

The first formulation introduced in [134] is based on the fact that many proximal algorithms, which rely on variable splitting techniques [137, 138, 139], require the computation of the proximal operator for the

regularizer. This is performed by solving a subproblem as follows:

$$\text{prox}_{\mu\Psi}(\boldsymbol{z}) := \underset{\boldsymbol{u}\in\mathbb{R}^n}{\arg\min}\frac{1}{2}\|\boldsymbol{u}-\boldsymbol{z}\|_2^2 + \mu\Psi(\boldsymbol{u}) \tag{4.8}$$

with $\mu > 0$ weighting the regularization.

The proximal operator is well-defined when the regularizer, Ψ, is a proper, lower semi-continuous convex function, ensuring that the minimum is unique. Moreover, for many non-smooth regularizers, the solutions to (4.8) can be derived through closed-form expressions, removing the need for explicit differentiation [140].

Furthermore, the subproblem can be considered as an image denoising problem of an image $\boldsymbol{z}$ affected by additive white Gaussian noise from a statistical modelling point of view. To formalise this mathematically, the operator (4.8) can be substituted by a general denoiser $\mathsf{D} : \mathbb{R}^n \to \mathbb{R}^n$ in the iterative algorithm, following:

$$\mathsf{D}(\boldsymbol{z}) \approx \underset{\boldsymbol{u}\in\mathbb{R}^n}{\arg\min}\frac{1}{2}\|\boldsymbol{u}-\boldsymbol{z}\|_2^2 + \mu\Psi(\boldsymbol{u})$$

The substitution limits the theoretical interpretation of the resulting model, in the sense that a generic denoiser may not always be seen as the proximal mapping of a specific function when it is not-expansive [141], resulting in the absence of an explicit objective function minimized during the Plug-and-Play iterations and, therefore, complicating theoretical analysis and making it difficult to adapt standard convergence guarantees.

These limitations have been addressed by some alternative assumptions on the properties of the denoiser, to guarantee the fixed-point convergence [142, 143].

Given that the regularizer is implicitly defined by the denoising process, the focus has been directed to the identification of an explicit objective function to minimize during the training, producing several types of denoisers, which can be grouped into these three following main categories:

Regularization by denoising (RED). In this kind of framework, the regularization term is defined in an explicit form as

$$\Psi(\boldsymbol{u}) = \frac{1}{2}\boldsymbol{u}^T\left(\boldsymbol{u} - \mathsf{D}(\boldsymbol{u})\right)$$

with D a generic denoiser.

In imaging, conceptually, this regularization tends to put a penalty on images which deviate from the natural image manifold and present artefacts. This is obtained by measuring the difference between the input and its residual after undergoing the denoising process.

The most interesting advantage of this method is represented by the ability to derive a practical expression for its gradient, demanding specific stringent conditions are met, such as local homogeneity, strong passivity, and the symmetry of the Jacobian. Following these assumptions, the gradient becomes the residual:

$$\nabla\Psi(\boldsymbol{u}) = \left(\boldsymbol{u} - \mathsf{D}(\boldsymbol{u})\right)$$

This framework has been proven to succeed in many inverse problems applied to imaging [144, 145]. However, several denoisers do not satisfy the local homogeneity to derive the residual gradient [146]. To address this problem, there have been proposed alternative formulations of RED to provide a stronger theoretical foundation and justify its effectiveness [146]. Nevertheless, several challenges are unsolved, as the conditions necessary for rigorous convergence analysis are often impractical or difficult to satisfy in practical scenarios.

Minimum Mean Square Estimator (MMSE). Another category is obtained by the minimization of the expected value of the loss function that defines the difference between the noisy image and a random

variable which represents the natural image. If the loss is chosen as the squared Euclidean norm, the Bayes estimator 4.2 that minimizes the mean squared error is defined pointwise as:

$$\mathsf{D}(\boldsymbol{z}) = \mathbb{E}[\boldsymbol{u}|\boldsymbol{z}] = \int_{\mathbb{R}^n} \boldsymbol{u} p(\boldsymbol{u}|\boldsymbol{z}) d\boldsymbol{u}$$

However, MMSE denoisers can be non-expansive [147], leading to the impossibility of computing the posterior distribution, which requires complex Monte Carlo sampling techniques for high-dimensional integration [148].

Gribonval formally established an explicit regularizer, demonstrating that the MMSE denoiser can be interpreted as the proximal operator of this regularizer [149]. Starting from this, numerous convergence results have been derived for this category of methods under general conditions [150].

Gradient Step Denoiser. In this last category, the gradient considered in the RED framework is derived using the gradient descent step realized on a potential function, $g : \mathbb{R}^n \to \mathbb{R}$, as follows:

$$\mathsf{D}(\boldsymbol{u}) = \boldsymbol{u} - \nabla g(\boldsymbol{u})$$

This denoiser is known as gradient step denoiser [151].

Under broad assumptions, such as ensuring that the gradient satisfies the contraction property, it can be shown that the class of denoisers are proximal operators [152]. This result is based on the characterization theorems of proximity operators outlined in [153]. The existence of the potential function such as $\mathsf{D}_\theta = \text{prox}_{\Psi_\theta}$, with its properties, can be used to analyse the convergence analysis of iterative Plug-and-Play algorithms [152].

Part II

Advanced Regularization and Learning for NMRD Analysis

Chapter 5

Regularization For Dipole-Dipole Relaxation

This chapter is based on the publication [154] where three potential approaches to analyse the NMRD profiles have been submitted:

- **MF-UPen**, which employs locally adapted L_2 regularization.
- **MF-L1**, an algorithm based on the L_1 penalty.
- **MF-MUPen**, that utilizes both locally adapted L_2 and global L_1 penalties.

In all these approaches the regularization parameters are computed through automatic procedures founded on the Balancing Principle (BP) [80] and the Uniform Penalty principle [40].

Two-dimensional time-domain NMR relaxometry techniques inspire all the algorithms. The locally adapted L_2 regularization was originally introduced by Bortolotti et al. in [40], where the regularization parameters are determined by applying the Uniform Penalty principle. The global L_1 regularization has been applied in [155] which addresses the more complex problem of data exhibiting spurious peaks caused by Quadrupolar Relaxation Enhancement, and it is described in the next Chapter 6. Finally, the coupling of locally adapted L_2 and global L_1 penalties was originally introduced by Bortolotti et al. in [22] for inverting two-dimensional NMR relaxation data and has been adapted to NMRD profiles.

The contributions of this work are the following:

- The implementation and experimental testing of the **MF-UPen** algorithm, featuring a novel rule for automatically computing the threshold parameter β_0.
- The implementation and experimental testing of the **MF-MUPen** algorithm.
- Development of a *dispersion analysis* procedure, enabling the determination of the existence range for estimated parameters.

The diversity of results achievable with different algorithms is shown, focusing on fit quality and correlation time distribution.

Following this introduction, the mathematical problem and numerical methods are detailed in sections 5.1 and 5.2, respectively. Section 5.3 then discusses the results from testing on two sets of NMRD profiles, each representing significant potential scenarios.

5.1 The Discrete Model for Dipole-Dipole Relaxation

As introduced in Section 1.2, spin relaxation theory represents the relaxation rates as linear combinations of spectral density functions of the motion modulating the interactions, i.e., Fourier transform of the time correlation function.

Depending on the sample under investigation, different spin dynamical interactions may occur. Generally, at low field, for spins-1/2 the dipole-dipole interaction is the most significant contributor to the relaxation process. In this chapter, let us focus on data acquired by FFC-NMR experiments from samples having spins-1/2. Hence, the contribution one has to take into account will be the dipole-dipole relaxation by ^1H nuclei.

Starting from the formulation of the model-free introduced by Conte et al. in [26], the NMRD profile, R_1, can be formulated as follows:

$$R_1(\omega) = R_0 + R^{HH}(\omega) \tag{5.1}$$

where ω is the angular frequency, R_0 is a non-negative offset keeping into account very *fast* molecular motions, and the term $R^{HH}(\omega)$ describes the correlation time distribution function $f(\tau)$ as follows:

$$R^{HH}(\omega) = \int_0^\infty \left[\frac{\tau}{(1+(\omega\tau)^2)} + \frac{4\tau}{(1+4(\omega\tau)^2)} \right] f(\tau) d\tau \tag{5.2}$$

where the correlation time τ is the average time required by a molecule to rotate one radiant or to move for a distance as large as its radius of gyration. The integral form described by (5.2) unconstrainedly retrieves only the number of possible correlation times representing the dynamics of the overall physical system. The other typical FFC data analysis approaches, on the contrary, rely on employing ad-hoc mathematical models containing information about both the number and meaning of the correlation times which describe the dynamic of a given system. Therefore, using (5.2) allows one to obtain a fingerprint of the possible motion regimes without the physical-chemical interpretation, which can be reasonably attempted further considering the chemistry of the sample under investigation.

The NMRD profile acquired by an FFC-NMR experiment is sampled at a finite number of angular frequencies. Let us introduce the following notation before describing the discretization of the continuous model (5.1). Let $\boldsymbol{\omega} \in \mathbb{R}^m$ be the vector of the m Larmor angular frequency values at which the profile R_1 is evaluated (with $\omega = 2\pi\nu$, and ν in [MHz]). Let $\boldsymbol{y} \in \mathbb{R}^m$ the vector which discretizes the $\boldsymbol{R}_1$, i.e., $y_i = R_1(\omega_i)$, with $i = 1, \ldots, m$. Finally, let $\boldsymbol{f} \in \mathbb{R}^n$ the vector obtained by sampling the correlation time distribution function $f(\tau)$ in n finite number logarithmically equispaced values $\tau_1, \ldots, \tau_n$.

By discretizing the equation (5.1), the following linear system is obtained:

$$\boldsymbol{y} = \mathcal{F}(\boldsymbol{f}, R_0) \equiv \mathcal{F}_1(\boldsymbol{f}) + R_0 \tag{5.3}$$

where $\mathcal{F} : \mathbb{R}^{n+1} \to \mathbb{R}^m$. Specifically, the first term, $\mathcal{F}_1 : \mathbb{R}^n \to \mathbb{R}^m$, is a linear function, depending only on $\boldsymbol{f}$, obtained by the discretization of the Fredholm integral equation (5.2), and can be expressed as:

$$\mathcal{F}_1(\boldsymbol{f}) \equiv \boldsymbol{K}\boldsymbol{f} \tag{5.4}$$

where the matrix $\boldsymbol{K} \in \mathbb{R}^{m \times n}$ represents the Lorentzian kernel (equation (1.24), section 1.3.5) as follows:

$$\boldsymbol{K}_{i,j} = \frac{\tau_j}{\left(1+(\omega_i\tau_j)^2\right)} + \frac{4\tau_j}{\left(1+4(\omega_i\tau_j)^2\right)} \quad , \qquad i = 1, \ldots, m \quad , \quad j = 1, \ldots, n \tag{5.5}$$

In a typical FFC-NMR experiment, the number of elements in R_1 is much smaller than the number of sampling values of the correlation time, i.e., $m \ll n$.

Finally, the last term in $\mathcal{F}$ of (5.3) is the constant parameter $R_0 \geq 0$, representing the offset in the NMRD profile.

Moreover, the previous equation (5.3) can be rewritten in a more compact form by introducing the following change of variables $\boldsymbol{x} \equiv (\boldsymbol{f}, R_0) \in \mathbb{R}^{n+1}$, and $\boldsymbol{K}_e = [\boldsymbol{K} \ \ \mathbf{1}] \in \mathbb{R}^{m\times(n+1)}$:

$$\boldsymbol{y} = \boldsymbol{K}_e \boldsymbol{x} \tag{5.6}$$

To address the problem of estimating the parameters $\boldsymbol{x}$, starting from $\boldsymbol{y}$, different solutions are presented in the next sections, representing one of the contributions of this thesis. The mathematical problem, being ill-conditioned, is reformulated as an inverse problem with specific regularization strategies to take into account some a priori information from physics and chemistry.

5.2 Numerical Methods Proposed

Recalling the notation from Chapter 2, for all three proposed algorithms, the fidelity term is the following

$$\Phi(\boldsymbol{x}, \boldsymbol{y}) = \|\boldsymbol{y} - \boldsymbol{K}_e \boldsymbol{x}\|_2^2$$

while, considering the regularization term, defined as $\Psi(x)$, different kinds of regularization strategies have been employed, following the theory discussed in Chapters 2, and 3.

Specifically, the *MF-UPen* algorithm is based on locally adapted L_2 regularization, the *MF-L1* algorithm is L_1-based regularization, and finally, *MF-MUPen* based on multi-penalty regularization, i.e., local-L_2, and L_1 penalties (described in 3.2).

5.2.1 MF-UPen Algorithm

This algorithm, implementing the locally adapted L_2 regularization, solves the following constrained minimization problem:

$$\min_{\boldsymbol{x}\geq 0} \left\{ \|\boldsymbol{y} - \boldsymbol{K}_e \boldsymbol{x}\|_2^2 + \sum_{i=1}^{n} \lambda_i \left(\boldsymbol{L}\boldsymbol{x}\right)_i^2 \right\} \tag{5.7}$$

with $\boldsymbol{L} = [\Delta, \mathbf{0}] \in \mathbb{R}^{n\times(n+1)}$ where Δ is the discretization of the second derivative operator, according to central finite difference formulas, and $\mathbf{0}$ is the $n-$components null column vector. Observe that the regularization is imposed only on the parameter $\boldsymbol{f}$ since the sum in (5.7) ranges for indices i from 1 to n. The regularization parameters λ_i, $i = 1, \ldots, n$ are computed according to the following relaxed UPEN principle (3.12) described in section 3.2 and introduced in [40]:

$$\lambda_i = \frac{\|\boldsymbol{y} - \boldsymbol{K}_e \boldsymbol{x}\|^2}{n\left(\beta_0 + \beta_p \max_{\mu\in I_i}\left(\boldsymbol{p}_\mu\right)^2 + \beta_c \max_{\mu\in I_i}\left(\boldsymbol{c}_\mu\right)^2\right)} \quad , \qquad i = 1, \ldots, n \tag{5.8}$$

where $\boldsymbol{c} = \boldsymbol{L}\boldsymbol{x}$, $\boldsymbol{p} = [\nabla, \mathbf{0}]\boldsymbol{x}$ and the I_i are the indices subsets related to the neighbourhood of the $i-$th entry, i.e. $I_i = \{i-1, i, i+1\}$. The β's are positive parameters. The parameter β_0 prevents division by zero and is a compliance floor, which should be small enough to prevent under-smoothing and large enough to avoid over-smoothing. The optimum value of β's could substantially change with the nature of the measured sample.

The parameters β_p, β_c are used to enhance or mitigate the local effects of slope or curvature. A preliminary trial value that often yields satisfactory results is $\beta_p = \beta_c = 1$. The parameter β_0, however, is more critical; its value should not exceed the threshold defined by

$$\max_i \left\{ \beta_p \max_{\mu\in I_i}\left(\boldsymbol{p}_\mu\right)^2 + \beta_c \max_{\mu\in I_i}\left(\boldsymbol{c}_\mu\right)^2 \right\} \tag{5.9}$$

while a too-small value, especially in cases where slope ($\boldsymbol{p}$) and curvature ($\boldsymbol{c}$) approach zero, would lead to an extremely ill-conditioned problem, hence causing computational challenges.

Therefore, an automatic rule for determining β_0 based on the estimate of (5.9) has been proposed, and obtained from a tentative solution $\widehat{\boldsymbol{f}}$ computed by the Truncated Singular Value Decomposition [156] of the matrix $\boldsymbol{K} = U\Sigma V^T$:

$$\widehat{\boldsymbol{f}} = \sum_{\sigma_i > \mathrm{Tol}_\sigma} \frac{\boldsymbol{u}_i^T y}{\sigma_i}\boldsymbol{v}_i \quad , \qquad \mathrm{Tol}_\sigma = 10^{-6}\sigma_1$$

where $\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_i \geq \cdots$ represent the singular values, and the vectors $\boldsymbol{u}_i, \boldsymbol{v}_i$ represent the i-th columns of U and V, respectively [157].

By setting:

$$V_i = \beta_p \max_{\mu \in I_i} \left(\widehat{\boldsymbol{p}}_\mu\right)^2 + \beta_c \max_{\mu \in I_i} \left(\widehat{\boldsymbol{c}}_\mu\right)^2 \quad , \qquad i = 1, \ldots, n,$$

where $\widehat{\boldsymbol{c}} = \Delta\widehat{\boldsymbol{f}}$ and $\widehat{\boldsymbol{p}} = \nabla\widehat{\boldsymbol{f}}$, the β_0 term can be derived as follows:

$$\beta_0 = \rho\|V\|_\infty \quad , \qquad 0 < \rho < 1. \tag{5.10}$$

The advantage of this approach is that it substitutes the parameter β_0, which can range in $(0, \infty)$, with the parameter ρ, which is confined within the interval $(0, 1)$. This substitution ensures that β_0 remains lower than the highest values of V but higher than the lowest ones. This makes determining β_0 more intuitive, particularly when supported by a visual representation of V.

Summarizing, *MF-UPen* is an iterative scheme where, given an initial guess $\lambda_i^{(0)}$, $i = 1, \ldots, n$, an approximate solution $\boldsymbol{x}^{(k)} \equiv (\boldsymbol{f}^{(k)}, R_0^{(k)})$ is computed by solving (5.7) for fixed $\lambda_i^{(k)}$, $i = 1, \ldots, n$, and the regularization parameters values are updated according to (5.8). The minimization problem (5.7) is solved by the Newton projection method (NP) [76].

MF-UPen is stated in the following Algorithm 6. The iterations are stopped when the following condition is satisfied

$$\sum_{i=1}^{n} |\lambda_i^{(k+1)} - \lambda_i^{(k)}| < \mathrm{Tol} \sum_{i=1}^{n} |\lambda_i^{(k)}| \tag{5.11}$$

where Tol is a fixed tolerance.

Algorithm 6 *MF-UPen*

1: Set $k = 0$, and choose a starting guess $\lambda_i^{(0)}$, $i = 1, \ldots, n$.
2: Compute β_0 according to (5.10).
3: **repeat**
4: $\quad k = k + 1$
5: $\quad$ **NMRD parameters update.** By using the Newton Projection method compute

$$\boldsymbol{x}^{(k)} = \arg\min_{\boldsymbol{x}} \|\boldsymbol{y} - \boldsymbol{K}_e\boldsymbol{x}\|_2^2 + \sum_{i=1}^{n} \lambda_i^{(k-1)} (\boldsymbol{L}\boldsymbol{x})_i^2$$

6: $\quad$ **Regularization parameter update.** Set

$$\lambda_i^{(k)} = \frac{\left\|\boldsymbol{y} - \boldsymbol{K}_e\boldsymbol{x}^{(k)}\right\|^2}{n\left(\beta_0 + \beta_p \max_{\mu \in I_i}\left(\boldsymbol{p}_\mu^{(k)}\right)^2 + \beta_c \max_{\mu \in I_i}\left(\boldsymbol{c}_\mu^{(k)}\right)^2\right)} \quad , \qquad i = 1, \ldots, n$$

7: **until** `converge condition` (5.11)
8: **return** $(\boldsymbol{f}, R_0) = \boldsymbol{x}^{(k)}$ ▷ Result $(\boldsymbol{f}, R_0)$

5.2.2 MF-L1 Algorithm

This algorithm employs an L_1-norm-based penalty, which is preferred for inducing sparsity in $\boldsymbol{f}$. This approach is based on the assumption that the $f(\tau)$ distribution is a sparse function characterized by only a few non-zero terms.

The problem of parameter identification is reformulated as the following optimization problem:

$$\min_{\boldsymbol{x}\geq 0}\left\{\|\boldsymbol{y}-\boldsymbol{K}_e\boldsymbol{x}\|_2^2+\alpha\|\boldsymbol{x}\|_1\right\} \tag{5.12}$$

where $\alpha>0$ is the regularization parameter computed according to the Balancing Principle (BP) introduced in Section 3.1.

Following [155], the equation (5.12) can be rewritten as

$$\begin{array}{ll}\min\limits_{\boldsymbol{x}} & \left\{\|\boldsymbol{y}-\boldsymbol{K}_e\boldsymbol{x}\|_2^2+\alpha\|\boldsymbol{x}\|_1+\eta\|\boldsymbol{x}\|_2^2\right\}\\ \text{s.t.} & \boldsymbol{x}\geq 0\end{array} \tag{5.13}$$

In this new formulation (5.13) the last L_2-based penalty term, $\eta\|\boldsymbol{x}\|_2^2$, has been introduced only to ensure that $\boldsymbol{K}_e^T\boldsymbol{K}_e+\eta\boldsymbol{I}$ is a definite positive matrix to ensure that (5.13) is well-posed. It is not a regularization term and a small positive value for $\eta\approx 10^{-10}$ is fixed. A complete discussion about the introduction of this term is in the next Chapter (6), and in [155].

The *MF-L1* algorithm is an iterative procedure where, starting from an initial guess $\lambda^{(0)}$, at each iteration k, an estimate of the parameters $(\boldsymbol{f}^{(k)},R_0^{(k)})$ is computed by solving the parameter estimation problem (5.13) for fixed $\alpha^{(k)}$, by the truncated Newton interior-point method [158] (see Algorithm 7, step (4)). Then a new value $\alpha^{(k+1)}$ is determined by using the BP (see Algorithm 7, step (5)). The BP selects the regularization parameter α so that the data fidelity and the regularization terms are balanced up to a multiplicative factor γ, i.e.,:

$$\gamma\alpha\|\boldsymbol{x}\|_1=\|\boldsymbol{y}-\boldsymbol{K}_e\boldsymbol{x}\|_2^2+\eta\|\boldsymbol{x}\|_2^2. \tag{5.14}$$

Using the value for $\gamma\ =1$ [85], the following rule for the parameter selection is obtained:

$$\alpha=\frac{\|\boldsymbol{y}-\boldsymbol{K}_e\boldsymbol{x}\|_2^2+\eta\|\boldsymbol{x}\|_2^2}{\|\boldsymbol{x}\|_1}. \tag{5.15}$$

The *MF-L1* method is summarized in the following Algorithm 7.

Algorithm 7 *MF-L1*

1: Set $k=0$, $\eta=10^{-10}$, and choose a starting guess $\alpha^{(0)}$.
2: **repeat**
3: $\quad k=k+1$
4: $\quad$ **NMRD parameters update.** By using the truncated Newton interior-point method, compute

$$\boldsymbol{x}^{(k)}=\arg\min_{\boldsymbol{x}\geq 0}\|\boldsymbol{y}-\boldsymbol{K}_e\boldsymbol{x}\|_2^2+\alpha^{(k-1)}\|\boldsymbol{x}\|_1+\eta\|\boldsymbol{x}\|_2^2$$

5: $\quad$ **Regularization parameter update.** Set

$$\alpha^{(k)}=\frac{\|\boldsymbol{y}-\boldsymbol{K}_e\boldsymbol{x}^{(k)}\|_2^2+\eta\|\boldsymbol{x}^{(k)}\|_2^2}{\|\boldsymbol{x}^{(k)}\|_1}$$

6: **until** $\|\alpha^{(k)}-\alpha^{(k-1)}|\leq \text{Tol}|\alpha^{(k)}|$
7: **return** $(\boldsymbol{f},R_0)=\boldsymbol{x}^{(k)}$ ▷ Result $(\boldsymbol{f},R_0)$

An extension of this algorithm, taking into account the QRE effect, is presented in the next Chapter 6.

5.2.3 MF-MUPen Algorithm

This last algorithm implements the multi-penalty approach, proposed in [40] for the two-dimensional NMR relaxometry data. *MF-MUPen* solves the following unconstrained minimization problem:

$$\min_{\boldsymbol{x}} \left\{ \|\boldsymbol{y} - \boldsymbol{K}_e\boldsymbol{x}\|_2^2 + \sum_{i=1}^{n} \lambda_i (\boldsymbol{L}\boldsymbol{x})_i^2 + \alpha \|\boldsymbol{x}\|_1 \right\} \tag{5.16}$$

which incorporates both penalty functions from *MF-UPen* and *MF-L1*. The regularization parameters are then calculated using (5.8) for $\lambda_i, i = 1, \ldots, n$ and (5.15) for α.

Summarizing, *MF-MUPen* is an iterative scheme where, given an initial guess $\lambda_i^{(0)}$, $i = 1, \ldots, n$, a parameter estimate $(\boldsymbol{f}^{(k)}, R_0^{(k)})$ is computed by solving (5.16) for fixed $\lambda_i^{(k)}$, $i = 1, \ldots, n+1$. Problem (5.16) is solved by the FISTA method proposed in [159] which is one of the most popular methods for minimizing L_1-penalized least squares functions.

MF-MUPen algorithm is stopped when the following condition is satisfied

$$\sum_{i=1}^{n} |\lambda_i^{(k+1)} - \lambda_i^{(k)}| + |\alpha^{(k+1)} - \alpha^{(k)}| < \text{Tol} \left(\sum_{i=1}^{n} |\lambda_i^{(k)}| + |\alpha^{(k)}| \right) \tag{5.17}$$

MF-MUPen is sketched in the following Algorithm 8.

Algorithm 8 *MF-MUPen*

1: Set $k = 0$, and choose a starting guess $\lambda_i^{(0)}$, $i = 1, \ldots, n$, $\alpha^{(0)}$.
2: **repeat**
3: $k = k + 1$
4: **NMRD parameters update.** By using the FISTA method, compute

$$\boldsymbol{x}^{(k)} = \arg\min_{\boldsymbol{x}} \|\boldsymbol{y} - \boldsymbol{K}_e\boldsymbol{x}\|_2^2 + \sum_{i=1}^{n} \lambda_i^{(k-1)} (\boldsymbol{L}\boldsymbol{x})_i^2 + \alpha^{(k-1)} \|\boldsymbol{x}\|_1$$

5: **Regularization parameter update.** Set

$$\lambda_i^{(k)} = \frac{\left\|\boldsymbol{y} - \boldsymbol{K}_e\boldsymbol{x}^{(k)}\right\|^2}{n\left(\beta_0 + \beta_p \max_{\mu \in I_i} \left(\boldsymbol{p}_\mu^{(k)}\right)^2 + \beta_c \max_{\mu \in I_i} \left(\boldsymbol{c}_\mu^{(k)}\right)^2\right)} \quad , \qquad i = 1, \ldots, n$$

$$\alpha^{(k)} = \frac{\left\|\boldsymbol{y} - \boldsymbol{K}_e\boldsymbol{x}^{(k)}\right\|^2}{n\|\boldsymbol{x}^{(k)}\|_1}$$

6: **until** `converge condition` (5.17)
7: **return** $(\boldsymbol{f}, R_0) = \boldsymbol{x}^{(k)}$ ▷ Result $(\boldsymbol{f}, R_0)$

These three Algorithms 6, 7 and 8 require an initial estimate for the regularization parameters. This can be obtained by computing a rough approximation $\tilde{\boldsymbol{x}}$ to the following nonnegatively constrained least squares problem

$$\min_{\boldsymbol{x} \geq 0} \|\boldsymbol{y} - \boldsymbol{K}_e\boldsymbol{x}\|_2^2$$

and then by using the Balancing and Uniform Penalty principles to get the initial guess. More precisely,

$$\alpha^{(0)} = \frac{\|\boldsymbol{y} - \boldsymbol{K}_e\tilde{\boldsymbol{x}}\|_2^2 + \eta\|\tilde{\boldsymbol{x}}\|_2^2}{\|\tilde{\boldsymbol{x}}\|_1}$$

in Algorithms 7, 8 and

$$\lambda_i^{(0)} = \frac{\|\boldsymbol{y} - \boldsymbol{K}_e\tilde{\boldsymbol{x}}\|^2}{n\left(\beta_0 + \beta_p \max_{\mu\in I_i}\left(\tilde{\boldsymbol{p}}_\mu\right)^2 + \beta_c \max_{\mu\in I_i}\left(\tilde{\boldsymbol{c}}_\mu\right)^2\right)} \quad , \qquad i = 1, \dots, n$$

in Algorithms 6 and 8.

5.3 Results and Discussion

This section reports and discusses the results obtained by the proposed algorithms on samples of two different materials that represent typical case tests.

In the first part, i.e., Section 5.3.1, the metrics to quantitatively evaluate the results' quality and the experimental setting are introduced; then, in the last part, i.e., Sections 5.3.2 and 5.3.3, the results obtained by the three algorithms are shown and discussed.

Numerical computations were carried out using Matlab R2022b on a laptop equipped with an Apple M1 processor with 16 GB of 2133 MHz RAM.

It should be noted that throughout the section, the frequencies ν are used instead of the angular frequencies ω, where $\nu \equiv \omega/(2\pi)$.

5.3.1 Experimental Setting

The fitted NMRD profiles, computed by Algorithms 6, 7 and 8 are compared to the R_1 data by means of the χ^2 value defined as follows:

$$\chi^2 = \sum_{i=1}^{m} \frac{(e_i - y_i)^2}{m-1} \tag{5.18}$$

where $\boldsymbol{e}$ is the estimated data value, i.e.

$$\boldsymbol{e} = \boldsymbol{K}\widetilde{\boldsymbol{f}} + \widetilde{R_0}$$

with $(\widetilde{\boldsymbol{f}}, \widetilde{R_0})$ the computed parameters.

The computed correlation time distributions, $\boldsymbol{f}$, derived by the three algorithms are quantitatively compared by determining the peak values and the area below the distribution in the neighbourhood of the correspondent peak, referring to this value as SpecificWeight. To describe this variable, let us introduce the following notation. Let us assume that f has n_p local maxima at the correlation times τ_{c_ℓ}, $\ell = 1, \dots, n_p$. Then let $\mathcal{I}_\ell$ be the neighbourhood of interest through the Full Width at Half Maximum parameter, as follows:

$$\mathcal{I}_\ell \equiv [\tau_\ell^{low}, \tau_\ell^{up}] \qquad \text{s.t.} \qquad f(\tau_\ell^{low}) = f(\tau_\ell^{up}) = \frac{1}{2} f(\tau_{c_\ell}) \qquad \ell = 1, \dots, n_p$$

Let the SpecificWeight metric represents the value for each peak τ_{c_ℓ}, such as:

$$\text{SpecificWeight}_\ell = \sum_{j=1}^{n_\ell} \tau_{c_j} f(\tau_{c_j}) \quad , \qquad \tau_{c_j} \in \mathcal{I}_\ell \tag{5.19}$$

where n_ℓ is the number of correlation times belonging to $\mathcal{I}_\ell$, $\ell = 1, \dots, n_p$.

The value of the tolerance parameters used in the stopping criteria of all algorithms is Tol $= 10^{-2}$. Moreover, a maximum number $k = 10$ of iterations has been set but never reached. The computational cost is evaluated in terms of execution time.

Finally, the algorithms' robustness has been tested by applying them to a set of $s = 500$ artificial profiles obtained by adding to R_1 uniformly distributed noise within the experimental error intervals. The purpose of these tests referred to as *dispersion analysis*, is to evaluate the intervals containing the recovered parameter R_0, and how the computed estimates scatter around the average value. Additionally, the aim is to examine how the position and value of the peaks vary in the recovered correlation times distributions.

5.3.2 Numerical results from FFC measures

The results obtained by applying all the algorithms to NMRD profiles acquired from two experimental samples, i.e., *Manganese* and *Poplar*, are presented in this section.

Both systems are considered a *gold standard* in relaxometry studies, especially when the involvement of paramagnetic species is necessary. The relaxometric properties of aqueous manganese solutions have been thoroughly investigated [160, 161], and as such, these solutions are routinely utilized to assess the performance and stability of instruments. Additionally, the characteristics of Poplar char have been extensively studied [162], making it an effective model for examining the textural properties and functional mobility of solvents within these porous materials. The NMRD profile for the manganese sample was acquired by the authors, while the data pertaining to Poplar char were taken from [163].

These two samples show how the algorithms' results can complement each other to improve the overall quality of the information provided. The global quality of the examined methods has been evaluated in terms of $\boldsymbol{\chi}^2$, offset R_0 and computation time.

The R_1 data for the Manganese Sample is measured at 26 frequency values ν, ranging within the interval $[10^{-2} - 10^{1}]$ MHz. The error intervals for these measurements vary from ± 0.4 to ± 1.1 $[s^{-1}]$. These are illustrated by the black error bars in the left panel of Figure 5.1.

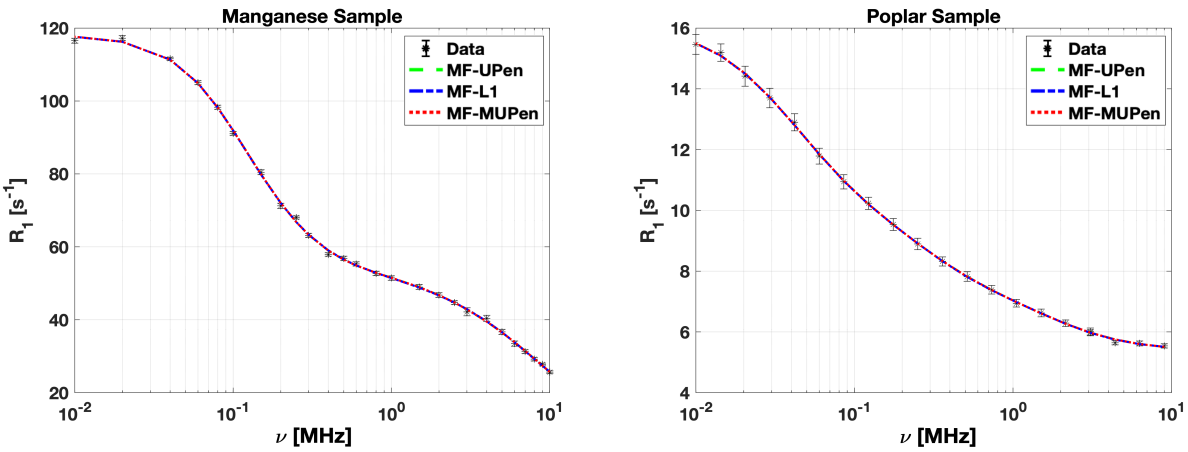


Figure 5.1: Comparison of R_1 relaxation rates for the Manganese and Poplar samples. The plots show the comparative analysis between the actual data (in black) and the results from the *MF-UPen* (in green), *MF-L1* (in blue), and *MF-MUPen* (in red) algorithms.

Table 5.1: Manganese Sample. Computational results of the proposed methods.

Algorithm	$\boldsymbol{R_0}$ $[s^{-1}]$	$\boldsymbol{\chi^2}$ $[-]$	**Computation time** $[s]$
MF-UPen	$6.64 \cdot 10^{0}$	$6.23 \cdot 10^{-1}$	$9.86 \cdot 10^{-1}$
MF-L1	$1.19 \cdot 10^{1}$	$5.55 \cdot 10^{-1}$	$1.35 \cdot 10^{-1}$
MF-MUPen	$9.98 \cdot 10^{0}$	$4.10 \cdot 10^{-1}$	$1.24 \cdot 10^{0}$

The Table 5.1 presents the estimated parameter R_0, the goodness-of-fit measure $\boldsymbol{\chi}^2$, and the computation time in seconds obtained by the three algorithms. The *MF-L1* algorithm achieves a moderate $\boldsymbol{\chi}^2$ value and the shortest computation time. In contrast, the *MF-UPen* algorithm shows a slightly higher $\boldsymbol{\chi}^2$ value and requires a longer computation time. Finally, the *MF-MUPen* algorithm achieves the best fit, indicated by

the lowest $\boldsymbol{\chi}^2$, suggesting superior model accuracy, albeit with a reasonable increase in computation time as a trade-off.

Considering the Poplar sample, the R_1 data is measured at 21 frequency values ν, within the interval $[10^{-2} - 10^{1}]$ MHz. The error intervals for these measurements vary from ± 0.06 to ± 0.3 $[s^{-1}]$. These are illustrated by the black error bar in the right panel of Figure 5.1.

Table 5.2: Poplar Sample. Computational results of the proposed methods.

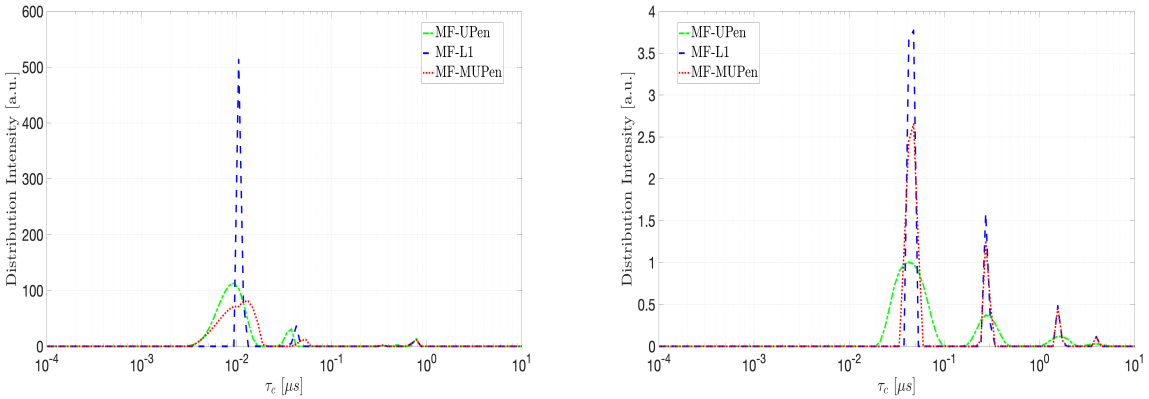
Algorithm	$\boldsymbol{R_0}$ $[s^{-1}]$	$\boldsymbol{\chi^2}$ $[-]$	**Computation time** $[s]$
MF-UPen	$5.40 \cdot 10^{0}$	$7.94 \cdot 10^{-3}$	$7.41 \cdot 10^{-2}$
MF-L1	$5.41 \cdot 10^{0}$	$8.84 \cdot 10^{-3}$	$6.65 \cdot 10^{-2}$
MF-MUPen	$5.41 \cdot 10^{0}$	$2.19 \cdot 10^{-2}$	$3.76 \cdot 10^{-1}$

The Table 5.2 outlines the computational results obtained for the Poplar sample by the three algorithms. Specifically, *MF-UPen* and *MF-L1* both report nearly identical values for R_0, with minimal $\boldsymbol{\chi}^2$ and very short computation times, indicating efficient and effective performance. However, *MF-MUPen*, while yielding a similar R_0 to the other two algorithms, shows a higher $\boldsymbol{\chi}^2$ value, suggesting a slightly poorer fit. Moreover, *MF-MUPen* requires longer computation time.

The outcomes for the Manganese and Poplar samples represent two scenarios, each indicative of the potential variability in sample analysis. This diversity highlights the importance of utilizing multiple methods to fully understand sample characteristics under varying conditions.

The peak analysis for both the Manganese and Poplar samples across the three methods is performed by plotting the correlation times amplitudes $\boldsymbol{f}$ computed by each method in Figure 5.2 and reporting in the Tables 5.3, 5.4 peaks positions amplitudes, half-width and SpecificWeights for each sample.

Considering the manganese peaks in Table 5.3, a perfect agreement is observed among the three methods in locating the peak at the longest correlation time, $\tau_c = 7.74 \cdot 10^{-1}$ μs. Meanwhile, *MF-UPen* and *MF-L1* show a quite good agreement at intermediate correlation times: $\tau_c = 3.76 \cdot 10^{-2}$ μs and $\tau_c = 4.23 \cdot 10^{-2}$ μs, respectively. The distribution pattern in Figure 5.2, left panel, shows similarity features between *MF-UPen* and *MF-MUPen* and reveals a tendency of *MF-L1* to resolve multiple components at the shortest times.



(a) Manganese sample.

(b) Poplar sample.

Figure 5.2: Distribution intensity as a function of τ_c. The plots show the results from the *MF-UPen* (in green), *MF-L1* (in blue), and *MF-MUPen* (in red) algorithms.

Table 5.3: Manganese Sample Analysis. Position (τ_c) and amplitude $f(\tau_c)$ of the distribution peaks sorted by $f(\tau_c)$.

Algorithm	τ_c $[\mu s]$	$f(\tau_c)$ $[a.u.]$	Half-width $[\mu s]$	SpecificWeight $[a.u.]$
MF-UPen	$8.30 \cdot 10^{-3}$	$9.78 \cdot 10^{1}$	$8.19 \cdot 10^{-3}$	$6.94 \cdot 10^{0}$
	$3.76 \cdot 10^{-2}$	$2.95 \cdot 10^{1}$	$8.47 \cdot 10^{-3}$	$2.13 \cdot 10^{0}$
	$\mathbf{7.74 \cdot 10^{-1}}$	$9.61 \cdot 10^{0}$	$1.18 \cdot 10^{-1}$	$1.09 \cdot 10^{1}$
	$4.86 \cdot 10^{-1}$	$4.63 \cdot 10^{0}$	$5.67 \cdot 10^{-2}$	$2.25 \cdot 10^{0}$
	$3.43 \cdot 10^{-1}$	$4.85 \cdot 10^{-2}$	$4.98 \cdot 10^{-2}$	$2.17 \cdot 10^{-2}$
MF-L1	$1.05 \cdot 10^{-2}$	$5.13 \cdot 10^{2}$	$1.34 \cdot 10^{-3}$	$6.30 \cdot 10^{0}$
	$4.23 \cdot 10^{-2}$	$3.63 \cdot 10^{1}$	$5.36 \cdot 10^{-3}$	$1.75 \cdot 10^{0}$
	$\mathbf{7.74 \cdot 10^{-1}}$	$1.16 \cdot 10^{1}$	$1.21 \cdot 10^{-1}$	$1.24 \cdot 10^{1}$
	$3.43 \cdot 10^{-1}$	$2.28 \cdot 10^{0}$	$4.00 \cdot 10^{-2}$	$7.84 \cdot 10^{-1}$
	$1.63 \cdot 10^{-3}$	$3.79 \cdot 10^{-3}$	$7.04 \cdot 10^{-4}$	$5.17 \cdot 10^{-5}$
MF-MUPen	$9.33 \cdot 10^{-3}$	$8.49 \cdot 10^{1}$	$1.09 \cdot 10^{-2}$	$6.98 \cdot 10^{0}$
	$4.75 \cdot 10^{-2}$	$1.09 \cdot 10^{1}$	$1.55 \cdot 10^{-2}$	$1.47 \cdot 10^{0}$
	$\mathbf{7.74 \cdot 10^{-1}}$	$1.15 \cdot 10^{1}$	$1.23 \cdot 10^{-1}$	$1.24 \cdot 10^{1}$
	$3.43 \cdot 10^{-1}$	$1.49 \cdot 10^{0}$	$5.24 \cdot 10^{-2}$	$7.56 \cdot 10^{-1}$

Table 5.4: Poplar Sample Analysis. Position (τ_c) and amplitude $f(\tau_c)$ of the distribution peaks sorted by $f(\tau_c)$.

Algorithm	τ_c $[\mu s]$	$f(\tau_c)$ $[a.u.]$	Half-width $[\mu s]$	SpecificWeight $[a.u.]$
MF-UPen	$4.23 \cdot 10^{-2}$	$1.02 \cdot 10^{0}$	$3.92 \cdot 10^{-2}$	$3.54 \cdot 10^{-1}$
	$3.05 \cdot 10^{-1}$	$4.65 \cdot 10^{-1}$	$1.25 \cdot 10^{-1}$	$5.00 \cdot 10^{-1}$
	$1.56 \cdot 10^{0}$	$1.75 \cdot 10^{-1}$	$5.38 \cdot 10^{-1}$	$8.05 \cdot 10^{-1}$
	$3.51 \cdot 10^{0}$	$5.53 \cdot 10^{-2}$	$9.32 \cdot 10^{-1}$	$4.53 \cdot 10^{-1}$
MF-L1	$4.75 \cdot 10^{-2}$	$3.77 \cdot 10^{0}$	$1.04 \cdot 10^{-2}$	$3.34 \cdot 10^{-1}$
	$2.72 \cdot 10^{-1}$	$1.59 \cdot 10^{0}$	$3.50 \cdot 10^{-2}$	$5.11 \cdot 10^{-1}$
	$1.56 \cdot 10^{0}$	$4.85 \cdot 10^{-1}$	$1.89 \cdot 10^{-1}$	$8.20 \cdot 10^{-1}$
	$3.94 \cdot 10^{0}$	$1.15 \cdot 10^{-1}$	$4.60 \cdot 10^{-1}$	$4.55 \cdot 10^{-1}$
MF-MUPen	$4.23 \cdot 10^{-2}$	$1.87 \cdot 10^{0}$	$2.13 \cdot 10^{-2}$	$3.40 \cdot 10^{-1}$
	$2.72 \cdot 10^{-1}$	$7.80 \cdot 10^{-1}$	$7.49 \cdot 10^{-2}$	$5.11 \cdot 10^{-1}$
	$1.56 \cdot 10^{0}$	$4.41 \cdot 10^{-1}$	$2.02 \cdot 10^{-1}$	$8.19 \cdot 10^{-1}$
	$3.94 \cdot 10^{0}$	$1.14 \cdot 10^{-1}$	$4.60 \cdot 10^{-1}$	$4.50 \cdot 10^{-1}$

In the case of the Poplar sample, as shown in the right panel of Figure 5.2 and Table 5.4, a tighter clustering of peaks is observed across the methods, particularly at the highest amplitude peak around $\tau_c = 4.23 \cdot 10^{-2}$ μs. This indicates that all three methods are in agreement regarding the main features of the Poplar sample's distribution.

5.3.3 Dispersion Analysis

The robustness of the methods is investigated through the dispersion analysis, described in 5.3.1. The boxplots in Figure 5.3 offer a comparative view of algorithmic performance on the two samples. Each boxplot outlines the algorithms' interquartile range (IQR) and median of χ^2 values.

Uniformity in the medians is observed for the Manganese sample, with outliers indicated by red plus symbols, suggesting occasional significant deviations for *MF-UPen*. The symmetry of the data is apparent from the lengths of the whiskers.

Conversely, the Poplar sample exhibits a tighter IQR for each algorithm, denoting less variability. Despite the close median values indicating consistent algorithmic performance, outliers for *MF-L1* reveal notable deviations in some cases.

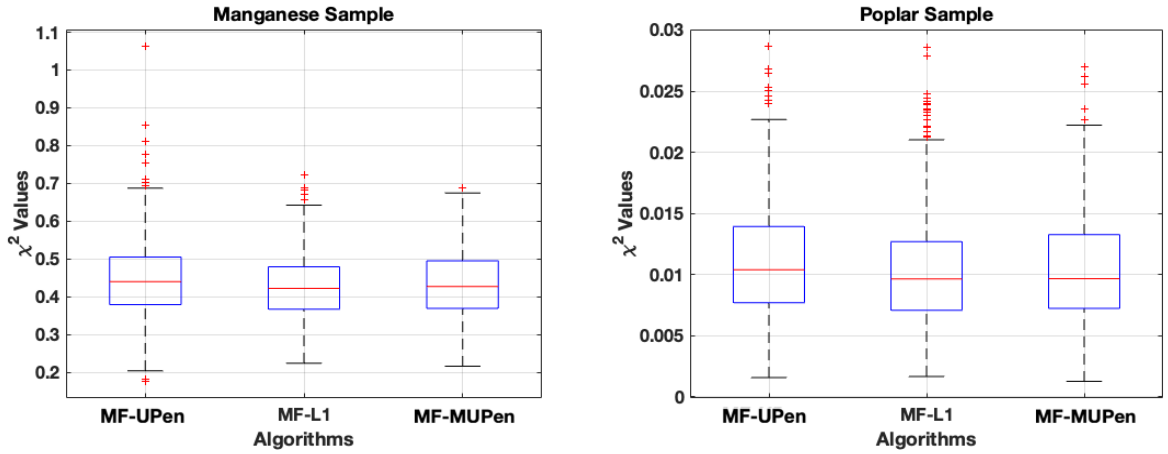


Figure 5.3: Boxplot of the χ^2 values for the Manganese and Poplar samples, comparing the results of the different algorithms on 500 data realizations.

Table 5.5 compares the R_0 confidence intervals [164], mean R_0, and medians for both Manganese and Poplar samples across the three algorithms.

The confidence intervals and mean R_0 values suggest a wider range of estimates for the Manganese sample, indicating a less uniform agreement among the algorithms. The median values, while closer, still reflect a notable variation between the algorithms, suggesting that the model fit depends on the algorithm applied.

Conversely, the Poplar sample demonstrates remarkable consistency, with both confidence intervals and mean R_0 values being nearly identical across all three algorithms. The median values also closely align, reinforcing the observation of uniform performance. This indicates that for the Poplar sample, the choice of algorithm does not significantly influence the outcome, and all three algorithms provide equivalent information.

Table 5.5 represents two distinct scenarios that may emerge when these algorithms are applied to samples with varying characteristics. In the case of the Poplar sample, the outcome from all three algorithms is congruent, implying that the algorithms are robust and interchangeable for this type of sample. Conversely, the Manganese sample demonstrates less consistency across the algorithms, suggesting that additional insights from alternative investigative methods are necessary to supplement the analysis.

Regarding the distribution intensities, the mean distribution obtained by each method was computed, and the peak positions and amplitudes were analysed in a manner analogous to Tables 5.1 and 5.4 for the single samples.

Table 5.5: Comparison of R_0 Confidence Intervals, Mean R_0, and Median for Manganese and Poplar samples.

Sample	**Algorithm**	R_0 Confidence Interval $[s^{-1}]$	R_0 Mean $[s^{-1}]$	Median $[s^{-1}]$
Manganese	*MF-UPen*	[5.240, 9.253]	$7.25 \cdot 10^0$	$8.26 \cdot 10^0$
	MF-L1	[9.251, 12.12]	$1.07 \cdot 10^1$	$1.12 \cdot 10^1$
	MF-MUPen	[9.652, 11.38]	$1.05 \cdot 10^1$	$1.05 \cdot 10^1$
Poplar	*MF-UPen*	[5.363, 5.406]	$5.39 \cdot 10^0$	$5.39 \cdot 10^0$
	MF-L1	[5.370, 5.413]	$5.39 \cdot 10^0$	$5.39 \cdot 10^0$
	MF-MUPen	[5.370, 5.416]	$5.39 \cdot 10^0$	$5.39 \cdot 10^0$

From Tables 5.6 and 5.7, it is observed that *MF-L1* identifies a greater number of peaks compared to the other two methods, indicating a higher sensitivity of the algorithm.

In the case of the Manganese sample, the data reported in Table 5.6 show that there is a perfect correspondence in peak position at the longest correlation time $\tau_c = 7.743 \cdot 10^{-1}$ μs among the three algorithms. While the peaks at shortest and intermediate times are split into multiple components.

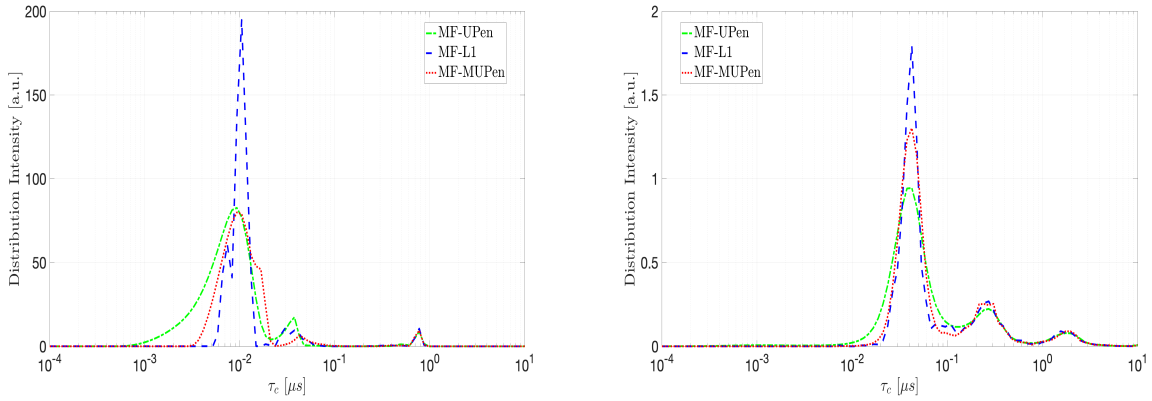
Concerning the Poplar sample (Table 5.7), it is observed that all algorithms exhibit identical peak positions corresponding to the largest amplitude, occurring at the shortest correlation time, $\tau_c = 4.229 \cdot 10^{-2}$ μs. At longest correlation times, *MF-UPen* finds a single peak around $\tau_c = 1.748 \cdot 10^0$ μs while *MF-L1* and *MF-MUPen* split the amplitudes in two peaks at $\tau_c = 1.556 \cdot 10^0, 1.963 \cdot 10^0$ μs and $\tau_c = 1.556 \cdot 10^0, 2.205 \cdot 10^0$ μs respectively.

However, despite the differences in the number of peaks identified, Figure 5.4 shows that all three algorithms exhibit a fundamental robustness in the localization of the positions of the highest peaks.

From Table 5.5, it is observed that *MF-MUPen* exhibits the smallest confidence intervals in both samples. Additionally, Figure 5.3 indicates that the number of outliers is smaller for *MF-MUPen* compared to the other methods. Combined with the observations from Figure 5.4, these results suggest a higher robustness of *MF-MUPen* compared to the other methods.

This consistency suggests the algorithms' effectiveness in capturing the primary characteristics of both samples, indicating their reliability in identifying the central features of the distributions regardless of sample differences.

In Appendix A, the scatter plots related to the computed R_0 and some plots of the computed profiles, and correlation time distributions are shown as examples.



(a) Manganese sample.

(b) Poplar sample.

Figure 5.4: Mean distribution amplitude over 500 data realizations. The plots show the results from the *MF-UPen* (in green), *MF-L1* (in blue), and *MF-MUPen* (in red) algorithms.

Table 5.6: Manganese Sample. Analysis of mean distribution over 500 realizations. Position (τ_c) and amplitude $f(\tau_c)$ of the distribution peaks sorted by $f(\tau_c)$.

Algorithm	τ_c $[\mu s]$	$f(\tau_c)$ $(a.u.)$	Half-width $[\mu s]$	SpecificWeight $[a.u.]$
MF-UPen	$9.326 \cdot 10^{-3}$	$8.640 \cdot 10^{1}$	$8.889 \cdot 10^{-3}$	$6.949 \cdot 10^{0}$
	$3.765 \cdot 10^{-2}$	$1.727 \cdot 10^{1}$	$9.075 \cdot 10^{-3}$	$1.617 \cdot 10^{0}$
	$\mathbf{7.743 \cdot 10^{-1}}$	$9.004 \cdot 10^{0}$	$1.385 \cdot 10^{-1}$	$1.168 \cdot 10^{1}$
	$5.462 \cdot 10^{-1}$	$9.534 \cdot 10^{-1}$	$1.139 \cdot 10^{-1}$	$4.464 \cdot 10^{0}$
	$5.995 \cdot 10^{-2}$	$7.407 \cdot 10^{-1}$	$7.643 \cdot 10^{-3}$	$1.208 \cdot 10^{-1}$
MF-L1	$1.048 \cdot 10^{-2}$	$1.642 \cdot 10^{2}$	$4.173 \cdot 10^{-3}$	$6.216 \cdot 10^{0}$
	$6.579 \cdot 10^{-3}$	$3.024 \cdot 10^{1}$	$4.493 \cdot 10^{-4}$	$5.120 \cdot 10^{-1}$
	$\mathbf{7.743 \cdot 10^{-1}}$	$1.061 \cdot 10^{1}$	$1.264 \cdot 10^{-1}$	$1.215 \cdot 10^{1}$
	$3.765 \cdot 10^{-2}$	$8.763 \cdot 10^{0}$	$1.897 \cdot 10^{-2}$	$1.702 \cdot 10^{0}$
	$3.854 \cdot 10^{-1}$	$5.470 \cdot 10^{-1}$	$7.072 \cdot 10^{-2}$	$5.642 \cdot 10^{-1}$
	$4.863 \cdot 10^{-1}$	$3.761 \cdot 10^{-1}$	$3.575 \cdot 10^{-2}$	$4.885 \cdot 10^{-1}$
	$2.420 \cdot 10^{-1}$	$1.233 \cdot 10^{-1}$	$1.700 \cdot 10^{-2}$	$7.589 \cdot 10^{-2}$
MF-MUPen	$1.048 \cdot 10^{-2}$	$8.461 \cdot 10^{1}$	$9.725 \cdot 10^{-3}$	$6.954 \cdot 10^{0}$
	$\mathbf{7.743 \cdot 10^{-1}}$	$9.675 \cdot 10^{0}$	$1.466 \cdot 10^{-1}$	$1.219 \cdot 10^{1}$
	$4.229 \cdot 10^{-2}$	$5.630 \cdot 10^{0}$	$2.526 \cdot 10^{-2}$	$2.163 \cdot 10^{0}$
	$4.863 \cdot 10^{-1}$	$4.064 \cdot 10^{-1}$	$1.402 \cdot 10^{-1}$	$9.117 \cdot 10^{-1}$

Table 5.7: Poplar Sample. Analysis of mean distribution over 500 realizations. Position (τ_c) and amplitude $f(\tau_c)$ of the distribution peaks sorted by $f(\tau_c)$.

Algorithm	τ_c $[\mu s]$	$f(\tau_c)$ $(a.u.)$	Half-width $[\mu s]$	SpecificWeight $[a.u.]$
MF-UPen	$\mathbf{4.229 \cdot 10^{-2}}$	$7.234 \cdot 10^{-1}$	$5.094 \cdot 10^{-2}$	$3.369 \cdot 10^{-1}$
	$2.719 \cdot 10^{-1}$	$1.830 \cdot 10^{-1}$	$8.540 \cdot 10^{-2}$	$2.547 \cdot 10^{-1}$
	$1.748 \cdot 10^{0}$	$7.503 \cdot 10^{-2}$	$1.211 \cdot 10^{0}$	$1.091 \cdot 10^{0}$
MF-L1	$\mathbf{4.229 \cdot 10^{-2}}$	$1.652 \cdot 10^{0}$	$1.794 \cdot 10^{-2}$	$2.811 \cdot 10^{-1}$
	$2.719 \cdot 10^{-1}$	$2.662 \cdot 10^{-1}$	$1.465 \cdot 10^{-1}$	$4.595 \cdot 10^{-1}$
	$8.498 \cdot 10^{-2}$	$1.744 \cdot 10^{-1}$	$2.187 \cdot 10^{-2}$	$6.473 \cdot 10^{-2}$
	$1.204 \cdot 10^{-1}$	$1.287 \cdot 10^{-1}$	$1.081 \cdot 10^{-2}$	$3.840 \cdot 10^{-2}$
	$1.963 \cdot 10^{0}$	$8.927 \cdot 10^{-2}$	$1.045 \cdot 10^{0}$	$1.007 \cdot 10^{0}$
	$1.556 \cdot 10^{0}$	$8.810 \cdot 10^{-2}$	$1.176 \cdot 10^{-1}$	$3.459 \cdot 10^{-1}$
	$6.136 \cdot 10^{-1}$	$2.873 \cdot 10^{-2}$	$7.136 \cdot 10^{-2}$	$4.805 \cdot 10^{-2}$
	$7.925 \cdot 10^{0}$	$1.298 \cdot 10^{-3}$	$5.589 \cdot 10^{-1}$	$2.304 \cdot 10^{-2}$
MF-MUPen	$\mathbf{4.229 \cdot 10^{-2}}$	$1.312 \cdot 10^{0}$	$2.518 \cdot 10^{-2}$	$3.009 \cdot 10^{-1}$
	$2.420 \cdot 10^{-1}$	$2.697 \cdot 10^{-1}$	$1.420 \cdot 10^{-1}$	$4.568 \cdot 10^{-1}$
	$1.072 \cdot 10^{-1}$	$1.047 \cdot 10^{-1}$	$2.110 \cdot 10^{-2}$	$5.096 \cdot 10^{-2}$
	$1.556 \cdot 10^{0}$	$7.936 \cdot 10^{-2}$	$1.204 \cdot 10^{0}$	$1.180 \cdot 10^{0}$
	$2.205 \cdot 10^{0}$	$7.194 \cdot 10^{-2}$	$1.217 \cdot 10^{-1}$	$2.996 \cdot 10^{-1}$
	$6.893 \cdot 10^{-1}$	$2.774 \cdot 10^{-2}$	$1.504 \cdot 10^{-1}$	$8.125 \cdot 10^{-2}$
	$4.431 \cdot 10^{0}$	$7.331 \cdot 10^{-3}$	$2.851 \cdot 10^{-1}$	$7.834 \cdot 10^{-2}$

Chapter 6

Regularization For Quadrupolar Relaxation Enhancement Effect

This chapter is based on the publication [155].

As described in Section 1.2, the presence of nuclei with spins greater than 1/2 in the sample under examination results in the presence of the Quadrupolar Relaxation Enhancement (QRE) effect which contributes to the relaxation process. This complex effect is described in Sections 1.2, and 1.3. Essentially, it refers to electric interactions between these nuclei and the electric fields nearby, due to their charge distributions interacting with the gradient of the electric field generated by the surrounding electron clouds. The presence of the QRE phenomenon is represented by the presence of local maxima or distinct peaks across the NMRD profile acquired, due to resonant phenomena. To formalize this, let us focus on the contribution by the interaction between hydrogen (^1H) and nitrogen (^{14}N), which, for example, is typically manifested in the dynamics of solid proteins [3, 4]. The position of the peaks depends on the quadrupole parameters which are determined by the electric field gradient tensor at the ^{14}N position. Hence, slight changes in the electronic structure around ^{14}N cause changes in the position and the shape of the quadrupole peaks. Therefore, the QRE represents a very sensitive fingerprint of molecular arrangement which has a wide range of applications in several domains, from the study of ionic liquids, proteins [4], food science [17, 16, 26, 7], to environmental science [5].

Despite the consistent literature about the modelling of relaxation rate R_1 of protons fluids within a confined environment (see Section 1.3) and applications of FFC-NMR (see for instance [165] and references therein), the study of a computational framework for the automatization of the FFC-NMR analysis was still missing.

Therefore, the analysis of the NMRD profiles requires the solution of a parameter identification problem dealing with the estimation of the offset term, and the correlation time distribution (as presented in the previous Chapter), but also of the QRE parameters when in the presence of the QRE effect.

In the present contribution, the parameter identification problem is formulated as a regularized non-linear least squares problem with box constraints and, a completely automatic strategy for its solution is proposed. In particular, the objective function contains a non-linear least squares term, imposing data consistency, and a L_1-based regularization term. An L_1-based regularization term, added to the L_2-data fitting term, promotes sparse solutions since it forces only some components to be non-null while, at the same time, it pushes all the other components to zero. Since the correlation time distribution function is known to be sparse, i.e., to have only a few non-null values, L_1-based regularization is a suitable choice compared to L_2-based regularization, which indeed promotes smooth solutions. The data-fitting and regularization terms are balanced by the regularization parameter, and the physical constraints on the unknown parameters lead

to bound constraints in the optimization problem.

The parameter identification problem crucially depends on the regularization parameter whose value has to be properly identified to perform a meaningful NMRD analysis. Therefore, the proposed mathematical model depends on several parameters: the NMRD parameters (i.e. the offset, the correlation time distribution), the QRE parameters, and the regularization parameter. The estimation of all these parameters is carried out by an iterative process where, at each iteration, the regularization parameter is computed according to a balancing principle (introduced in Section 3.1) [80]. The NMRD and QRE parameters are estimated by solving the corresponding constrained optimization problem by the constrained two-blocks non-linear Gauss-Seidel (GS) method (presented in Section 2.2.3) [39, 38] since the unknown NMRD and QRE parameters can be naturally partitioned into two blocks. In the GS method, the objective function is iteratively minimized with respect to the offset and the correlation time distribution while the QRE parameters are held fixed; then, fixed the updated values for the offset and the correlation time distribution, the objective is minimized with respect to the QRE parameters. The first subproblem involves solving a constrained linear least squares problem, obtained by the *model-free* approach [26], with an L_1 regularization term. The second subproblem requires the solution of a constrained non-linear least squares problem.

This computational approach, separating the contribution due to the offset and the relaxation distributions from the parameters of the quadrupolar relaxation, can provide a very accurate fit not only of the overall NMRD profile but also of the local maxima due to the QRE.

Besides analyzing the convergence of the proposed approach, the method was on synthetic and real data aiming to illustrate its efficiency and robustness to data noise.

The remainder of this chapter is organized as follows: in Section 6.1 the mathematical model, its discretization, and the parameter identification problem are described; in Section 6.2 the solution method is introduced, and its properties are analyzed presenting the *AURORA* algorithm. The results from several numerical experiments are reported and discussed in Section 6.3.

6.1 The Discrete Model for QRE Effect

Following the model-free approach introduced in [26], let us now consider the case in the presence of the QRE effect, represented by the non-linear term R^{HN} to add in the equation (5.1) as follows:

$$R_1(\omega) = R_0 + R^{HH}(\omega) + R^{HN}(\omega) \tag{6.1}$$

For convenience, let us recall all the variables: the non-negative offset R_0 takes into account very fast molecular motions, the variable ω is the angular frequency, and the term $R^{HH}(\omega)$ is the linear term defining the correlation time distribution $f(\tau)$ described by the following integral:

$$R^{HH}(\omega) = \int_0^\infty \left[\frac{\tau}{(1+(\omega\tau)^2)} + \frac{4\tau}{(1+4(\omega\tau)^2)} \right] f(\tau) d\tau \tag{6.2}$$

Let us also recall that the integral form represented by (6.2) freely identifies the number of potential correlation times that characterize the overall dynamics of the physical system. In contrast, conventional FFC data analysis methods typically depend on pre-defined mathematical models, which incorporate assumptions about both the number and the interpretation of the correlation times that describe the system's dynamics. Therefore, applying (6.2) provides a broad fingerprint of the potential motion regimes without immediately assigning a physical-chemical meaning. This interpretation can later be refined by considering the specific chemical properties of the sample under study.

Finally, the last term $R^{HN}(\omega)$ represents the QRE phenomenon describing the occurrence of the *quadrupolar peaks* and it follows:

$$R^{HN}(\omega) = C^{HN}\left(\ \tfrac{1}{3} + \sin^2(\Theta)\cos^2(\Phi),\quad \tfrac{1}{3} + \sin^2(\Theta)\sin^2(\Phi),\quad \tfrac{1}{3} + \cos^2(\Theta)\ \right)\cdot \begin{pmatrix} \frac{\tau_Q}{1+(\omega-\omega_-)^2\tau_Q^2} + \frac{\tau_Q}{1+(\omega+\omega_-)^2\tau_Q^2} \\ \frac{\tau_Q}{1+(\omega-\omega_+)^2\tau_Q^2} + \frac{\tau_Q}{1+(\omega+\omega_+)^2\tau_Q^2} \\ \frac{\tau_Q}{1+(\omega-\Delta\omega)^2\tau_Q^2} + \frac{\tau_Q}{1+(\omega+\Delta\omega)^2\tau_Q^2} \end{pmatrix} \tag{6.3}$$

remarking that $\cdot$ operator in (6.3) denotes the scalar product between two vectors. The term R^{HN} depends on six parameters, detailed as follows:

i) C^{HN} refers to the gyromagnetic ratios and the average interaction distance of the nuclei;

ii) Θ and Φ are two angles accounting for the orientation of the $^1H - ^{14}N$ dipole-dipole axis with respect to the principal axis system of the electric field gradient at the position of ^{14}N;

iii) τ_Q is the correlation time for the $^1H - ^{14}N$ quadrupolar interaction;

iv) ω_- and ω_+ are the angular frequency position of the peaks on the NMRD profiles (where $\Delta\omega$ is the angular frequency difference $\omega_+ - \omega_-$).

Interpolating experimentally acquired data by using the model equation (6.1) can be considered as the combination of the free model approach given by (6.2) for the homonuclear linear R^{HH} term, with the non-linear function developed in the work [3], represented in (6.3) for the quadrupolar R^{HN} term.

In order to discretize this model, the same notation of the previous chapter is used, i.e., $\boldsymbol{\omega} \in \mathbb{R}^m$ is the vector of the m Larmor angular frequency values at which the profile R_1 is evaluated (with $\omega = 2\pi\nu$, and ν in [MHz]); the vector $\boldsymbol{y} \in \mathbb{R}^m$ discretizes the $\boldsymbol{R}_1$, i.e., $y_i = R_1(\omega_i)$, with $i = 1, \ldots, m$; and $\boldsymbol{f} \in \mathbb{R}^n$ represents the vector obtained by sampling the correlation time distribution function $f(\boldsymbol{\tau})$ in n finite number logarithmically equispaced values $\tau_1, \ldots, \tau_n$. Moreover, let $\boldsymbol{\psi} \in \mathbb{R}^6$ the vector collecting all the quadrupolar parameters, defined as follows:

$$\boldsymbol{\psi} \equiv (\psi_1, \psi_2, \psi_3, \psi_4, \psi_5, \psi_6) \equiv \left(C^{HN}, \sin^2(\Theta), \sin^2(\Phi), \tau_Q, \omega_-, \omega_+\right) \tag{6.4}$$

By discretizing the continuous model (6.1), the discrete model derived is the following:

$$\boldsymbol{y} = \mathcal{F}(\boldsymbol{f}, \boldsymbol{\psi}, R_0) \equiv \mathcal{F}_1(\boldsymbol{f}) + \mathcal{F}_2(\boldsymbol{\psi}) + R_0 \tag{6.5}$$

where $\mathcal{F} : \mathbb{R}^{n+6+1} \to \mathbb{R}^m$.

The first term has been introduced in the previous chapter in (5.4), and it is the linear function depending only on $\boldsymbol{f}$ obtained by the discretization of the Fredholm integral equation (6.2):

$$\mathcal{F}_1(\boldsymbol{f}) \equiv \boldsymbol{K}\boldsymbol{f} \tag{6.6}$$

where the matrix $\boldsymbol{K} \in \mathbb{R}^{m\times n}$ is defined as follows:

$$\boldsymbol{K}_{i,j} = \frac{\tau_j}{\left(1 + (\omega_i\tau_j)^2\right)} + \frac{4\tau_j}{\left(1 + 4\,(\omega_i\tau_j)^2\right)} \quad , \qquad i = 1, \ldots, m \quad , \quad j = 1, \ldots, n \tag{6.7}$$

Recalling that in a typical FFC-NMR experiment $m \ll n$.

The second term, $\mathcal{F}_2(\boldsymbol{\psi}) : \mathbb{R}^6 \to \mathbb{R}^m$ represents the discretization of the quadrupolar component R^{HN} (6.3), and it only depends on the quadrupolar parameters ψ_j, $j = 1, \ldots, 6$. It can be expressed as follows:

$$(\mathcal{F}_2(\boldsymbol{\psi}))_i = \psi_1 \left(\tfrac{1}{3} + \psi_2(1-\psi_3), \quad \tfrac{1}{3} + \psi_2 \cdot \psi_3, \quad \tfrac{1}{3} + (1-\psi_2) \right) \cdot \begin{pmatrix} \frac{\psi_4}{1+(\omega_i-\psi_5)^2\psi_4^2} + \frac{\psi_4}{1+(\omega_i+\psi_5)^2\psi_4^2} \\ \frac{\psi_4}{1+(\omega_i-\psi_6)^2\psi_4^2} + \frac{\psi_4}{1+(\omega_i+\psi_6)^2\psi_4^2} \\ \frac{\psi_4}{1+(\omega_i-(\psi_6-\psi_5))^2\psi_4^2} + \frac{\psi_4}{1+(\omega_i+(\psi_6-\psi_5))^2\psi_4^2} \end{pmatrix} \tag{6.8}$$

for $i = 1, \ldots, m$.

Finally, the last term in $\mathcal{F}$ is the constant parameter $R_0 \geq 0$, representing the offset in the NMRD profile.

The difference compared to the case presented in Chapter 5 is essentially related to introducing the non-linear term. Nevertheless, as for the correlation time distribution $\boldsymbol{f}$, it is typically possible to derive a priori information from the physics and the chemistry of the sample related to the quadrupolar parameters one wants to extract in this case, represented by the boundary set of $\boldsymbol{\psi}$:

$$\mathcal{B}_{\boldsymbol{\psi}} = \left\{ \boldsymbol{\psi} : \psi_1 \in [0, \bar{C}]; \psi_2, \psi_3 \in [0,1]; \psi_4 \in [0, \bar{\tau}]; \psi_5, \psi_6 \in [\omega_l, \omega_u] \right\} \tag{6.9}$$

A deeper discussion on this topic is in the next Section 6.3.1.

In this case, to stabilize the parameter identification procedure, knowing (6.1) is an ill-conditioned inverse problem, the L_1 regularization has been used to induce sparsity of $\boldsymbol{f}$, since the distribution $\boldsymbol{f}(\tau)$ is known to be a sparse function with only a few non-null terms. Therefore, the parameter identification problem is reformulated as the following optimization problem:

$$\begin{aligned} \min_{\boldsymbol{f}, \boldsymbol{\psi}, R_0} \quad & \| \boldsymbol{y} - (\mathcal{F}_1(\boldsymbol{f}) + \mathcal{F}_2(\boldsymbol{\psi}) + R_0) \|_2^2 + \lambda \|\boldsymbol{f}\|_1 \\ \text{s.t.} \quad & \boldsymbol{f} \geq \boldsymbol{0} \\ & \boldsymbol{\psi} \in \mathcal{B}_{\boldsymbol{\psi}} \\ & R_0 \geq 0 \end{aligned} \tag{6.10}$$

where, the first term of (6.10) represents the fidelity term, Φ, of Chapter 2; while the second term of (6.10) is the L_1 regularization term, Ψ, and it is weighted by the regularization parameter $\lambda > 0$. The parameters $(\boldsymbol{f}, \boldsymbol{\psi}, R_0)$ obtained by solving (6.10) depend critically on the value of λ.

6.2 Numerical Method

The presented parameter identification method is an iterative procedure where, at each iteration, a value of the regularization parameter λ is provided and the corresponding parameters $(\boldsymbol{f}_\lambda, \boldsymbol{\psi}_\lambda, R_{0,\lambda})$ are computed by solving problem (6.10). The constrained two-block non-linear Gauss-Seidel (GS) method (described in [38, 39], and here in Section 2.2.3) is used for its solution. In the following, first, for useful purposes, the GS method is recalled with its convergence properties, then, the iterative procedure for the regularization parameter computation is introduced, and, finally, the overall parameter identification procedure is drawn.

6.2.1 The constrained two-blocks Gauss-Seidel method

In this subsection, the GS method used for the solution of the constrained optimization problem (6.10) for a fixed value of the regularization parameter λ is described. To this end, the unknowns in (6.10) are partitioned into two blocks, such that the data fitting term becomes linear with respect to the first block and non-linear with respect to the second block. Consequently, problem (6.10) is reformulated as follows:

$$\begin{aligned} \min_{\boldsymbol{x}_1, \boldsymbol{x}_2} \quad & g(\boldsymbol{x}_1, \boldsymbol{x}_2) = \|\boldsymbol{y} - \boldsymbol{K}_e \boldsymbol{x}_1 - \mathcal{F}_2(\boldsymbol{x}_2)\|_2^2 + \lambda \|\boldsymbol{x}_1\|_1 + \eta \|\boldsymbol{x}_1\|_2^2 \\ \text{s.t.} \quad & \boldsymbol{x}_1 \in X_1 \\ & \boldsymbol{x}_2 \in X_2 \end{aligned} \tag{6.11}$$

where, as in the previous case, let $\boldsymbol{x}_1 \equiv (\boldsymbol{f}, R_0) \in X_1$ be the linear vector collecting the correlation time distribution vector and the constant offset, and $X_1 = \{\boldsymbol{x}_1 \geq 0\}$; while $\boldsymbol{x}_2 \equiv \boldsymbol{\psi} \in X_2$, with $X_2 \equiv \mathcal{B}_{\boldsymbol{\psi}}$. Moreover, let us recall $\boldsymbol{K}_e = [\boldsymbol{K} \;\; \mathbf{1}] \in \mathbb{R}^{m\times(n+1)}$.

The last L_2-based penalty term $\eta\|\boldsymbol{x}_1\|_2^2$ in the objective function has been introduced to ensure that $\boldsymbol{K}_e^T\boldsymbol{K}_e + \eta\boldsymbol{I}$ is positive definite; to this and, a small value for η, as $\eta = 10^{-10}$ for example, can be fixed. Moreover, observe that in (6.11), the parameter R_0 has been included in the L_1-based penalty term.

The closed subsets $X_1 \subseteq \mathbb{R}^{n+1}$ and $X_2 \subseteq \mathbb{R}^6$ are both convex; the objective function $g(\boldsymbol{x}_1, \boldsymbol{x}_2)$ is continuous and it is convex with respect to $\boldsymbol{x}_1$ for fixed $\boldsymbol{x}_2$, but it is not convex with respect to $\boldsymbol{x}_2$ for fixed $\boldsymbol{x}_1$. However, since $\boldsymbol{K}_e^T\boldsymbol{K}_e + \eta\mathbf{I}$ is positive definite and X_2 is bounded, it is easy to show that g is coercive on $X_1 \times X_2$.

Definition 6.1 *A function $g : \mathbb{R}^q \to \mathbb{R}$ is called coercive in X if, for every sequence $\{\boldsymbol{x}^{(k)}\} \in X$ such that $\|\boldsymbol{x}^{(k)}\| \to \infty$, we have*

$$\lim_{k\to\infty} g(\boldsymbol{x}^{(k)}) = +\infty$$

Proposition 6.1 *The function $g : \mathbb{R}^{n+1+6} \to \mathbb{R}$ such that*

$$g(\boldsymbol{x}_1, \boldsymbol{x}_2) = \|\boldsymbol{y} - \boldsymbol{K}_e\boldsymbol{x}_1 - \mathcal{F}_2(\boldsymbol{x}_2)\|_2^2 + \lambda\|\boldsymbol{x}_1\|_1 + \eta\|\boldsymbol{x}_1\|_2^2$$

is coercive in $X_1 \times X_2$.

Proof. The function g can be rewritten as

$$g(\boldsymbol{x}_1, \boldsymbol{x}_2) = \boldsymbol{x}_1^T(\boldsymbol{K}_e^T\boldsymbol{K}_e + \eta\boldsymbol{I})\boldsymbol{x}_1 + 2\boldsymbol{x}_1^T\boldsymbol{K}_e^T(\mathcal{F}_2(\boldsymbol{x}_2) - \boldsymbol{y}) + \|\mathcal{F}_2(\boldsymbol{x}_2) - \boldsymbol{y}\|^2 + \lambda\|\boldsymbol{x}_1\|_1$$

where $\boldsymbol{K}_e^T\boldsymbol{K}_e + \eta\boldsymbol{I}$ is positive definite. Let $\{(\boldsymbol{x}_1^{(k)}, \boldsymbol{x}_2^{(k)})\}$ be a sequence in $X_1 \times X_2$ such that $\lim_{k\to\infty}\|(\boldsymbol{x}_1^{(k)}, \boldsymbol{x}_2^{(k)})\| = \infty$. Since X_2 is bounded, we have

$$\lim_{k\to\infty}\|\boldsymbol{x}_1^{(k)}\| = \infty \qquad \text{and} \qquad \lim_{k\to\infty}\|\boldsymbol{x}_2^{(k)}\| < \infty \tag{6.12}$$

Let $\mu > 0$ be the smallest eigenvalue of $\boldsymbol{K}_e^T\boldsymbol{K}_e + \eta\boldsymbol{I}$. It holds

$$\begin{aligned} g(\boldsymbol{x}_1^{(k)}, \boldsymbol{x}_2^{(k)}) &\geq \mu\|\boldsymbol{x}_1^{(k)}\|^2 - 2\|\boldsymbol{K}_e^T(\mathcal{F}_2(\boldsymbol{x}_2^{(k)}) - \boldsymbol{y})\|\|\boldsymbol{x}_1^{(k)}\| + \lambda\|\boldsymbol{x}_1^{(k)}\| + \|\boldsymbol{K}_e^T(\mathcal{F}_2(\boldsymbol{x}_2^{(k)}) - \boldsymbol{y})\|^2 \\ &\geq \left(\mu\|\boldsymbol{x}_1^{(k)}\| - 2\|\boldsymbol{K}_e^T(\mathcal{F}_2(\boldsymbol{x}_2^{(k)}) - \boldsymbol{y})\| + \lambda\right)\|\boldsymbol{x}_1^{(k)}\| \end{aligned}$$

From (6.12), it follows that $\mu\|\boldsymbol{x}_1^{(k)}\| - 2\|\boldsymbol{K}_e^T(\mathcal{F}_2(\boldsymbol{x}_2^{(k)}) - \boldsymbol{y})\| + \lambda > 0$ for sufficiently large k. Therefore,

$$\lim_{k\to\infty} g(\boldsymbol{x}_1^{(k)}, \boldsymbol{x}_2^{(k)}) = +\infty$$

□

Continuity and coerciveness ensure the existence of at least one global minimizer of $g(\boldsymbol{x}_1, \boldsymbol{x}_2)$ in $X_1 \times X_2$ [76].

In the constrained two-blocks Gauss-Seidel method, at each iteration, the objective function is minimized with respect to each of the block coordinate vectors $\boldsymbol{x}_i$ over the subsets X_i, $i = 1, 2$, as recalled in the Algorithm 9, where the convergence condition is:

$$|g(\boldsymbol{x}_1^{(k)}, \boldsymbol{x}_2^{(k)}) - g(\boldsymbol{x}_1^{(k-1)}, \boldsymbol{x}_2^{(k-1)})| \leq \mathrm{Tol}_{GS}|g(\boldsymbol{x}_1^{(k)}, \boldsymbol{x}_2^{(k)})| \tag{6.13}$$

It is noted that the GS method is well-defined since each subproblem has solutions. Indeed, the function g is strictly convex with respect to $\boldsymbol{x}_1$ and hence there exists at most one global minimum of f over X_1 for

Algorithm 9 Constrained two-blocks non-linear Gauss-Seidel method

1: **function** GS($\boldsymbol{x}_1^{(0)}, \boldsymbol{x}_2^{(0)}$)
2: Set $k = 0$ and $\boldsymbol{x}^{(0)} = (\boldsymbol{x}_1^{(0)}, \boldsymbol{x}_2^{(0)})$
3: **repeat**
4: $k = k + 1$
5: Set $\boldsymbol{x}_1^{(k)} \in \arg\min_{\boldsymbol{z} \in X_1} g(\boldsymbol{z}, \boldsymbol{x}_2^{(k-1)})$
6: Set $\boldsymbol{x}_2^{(k)} \in \arg\min_{\boldsymbol{z} \in X_2} g(\boldsymbol{x}_1^{(k)}, \boldsymbol{z})$
7: **until** `convergence condition` (6.13)
8: **return** $(\boldsymbol{x}_1^{(k)}, \boldsymbol{x}_2^{(k)})$
9: **end function**

fixed $\boldsymbol{x}_2$. On the other hand, Weierstrass's theorem guarantees the existence of at least one global minimum of g over X_2 for fixed $\boldsymbol{x}_1$ since g is continuous and X_2 is a closed and bounded set.

For general nonconvex, constrained problems, the convergence of sequences generated by the GS method to critical points has been proved in [39]. For the reader's convenience, the main convergence result for the GS method is reported, referring to [39] for its proof.

Theorem 6.1 *Consider the problem*

$$\begin{aligned} \min_{\boldsymbol{x}_1, \boldsymbol{x}_2} \quad & g(\boldsymbol{x}_1, \boldsymbol{x}_2) \\ s.t. \quad & \boldsymbol{x}_1 \in X_1 \\ & \boldsymbol{x}_2 \in X_2 \end{aligned} \tag{6.14}$$

where g is a continuously differentiable function and the subsets X_i are closed, nonempty and convex for $i = 1, 2$. Suppose that the sequence $\{(\boldsymbol{x}_1^{(k)}, \boldsymbol{x}_2^{(k)})\}$ generated by the two-blocks GS method has limit points. Then, every limit point of $\{(\boldsymbol{x}_1^{(k)}, \boldsymbol{x}_2^{(k)})\}$ is a critical point of the problem.

It has already been observed that the objective function g in (6.11) is coercive; since the level sets of continuous coercive functions are compact, the sequence $\{(\boldsymbol{x}_1^{(k)}, \boldsymbol{x}_2^{(k)})\}$ generated by the GS method has limit points (eventually, it has a convergent subsequence); hence, the GS method converges to critical points of (6.11).

This subsection concludes with a remark regarding the solution of the two constrained subproblems that must be addressed at each iteration of Algorithm 9. The first subproblem at step 5 is an L_1-regularized least squares problem with nonnegativity constraints:

$$\begin{aligned} \min_{\boldsymbol{z}} \quad & \|\boldsymbol{\omega} - \boldsymbol{K}_e \boldsymbol{z}\|_2^2 + \lambda \sum_{i=1}^{m+1} z_i \\ \text{s.t.} \quad & z_i \geq 0 \quad , \qquad i = 1, \ldots, m+1 \end{aligned} \tag{6.15}$$

where $\boldsymbol{\omega} = \boldsymbol{y} - \mathcal{F}_2(\boldsymbol{x}_2^{(k)})$. For its solution, the truncated Newton interior-point method has been used described in [158].

The second subproblem in step 6 is a bound-constrained non-linear least squares problem:

$$\begin{aligned} \min_{\boldsymbol{z}} \quad & \|\mathcal{F}_2(\boldsymbol{z}) - \boldsymbol{\omega}\|^2 \\ \text{s.t.} \quad & \boldsymbol{z} \in X_2 \end{aligned} \tag{6.16}$$

where $\boldsymbol{\omega} = \boldsymbol{K}_e \boldsymbol{x}_1^{(k+1)} - \boldsymbol{y}$. For its solution, the Newton Projection method has been used [67, 68] where the Hessian matrix is approximated as in the Levenberg-Marquardt method [166] since the Jacobian of $\mathcal{F}_2$ is ill-conditioned.

6.2.2 Computation of the regularization parameter λ

To correctly analyze the NMRD profiles, it is necessary to choose an appropriate value for the regularization parameter λ. Even if several parameter selection rules have been proposed in the literature for L_2-regularized minimization problems, the case of L_1-based regularization remains largely unexplored (see Chapter 3). In [78, 79], the discrepancy principle has been investigated for nonsmooth regularization. This principle is difficult to be realized since it requires prior knowledge of the noise norm and a solution of the discrepancy equation is not guaranteed to exist. In [80], the Balancing Principle (BP) has been proposed where the regularization parameter is selected by balancing, up to a multiplicative factor γ, the data fidelity, and the regularization term, i.e.,

$$\gamma\lambda\|\boldsymbol{x}_1\|_1 = \|\boldsymbol{y} - \boldsymbol{K}_e\boldsymbol{x}_1 - \mathcal{F}_2(\boldsymbol{x}_2)\|_2^2 + \eta\|\boldsymbol{x}_1\|_2^2 \tag{6.17}$$

The regularization properties of the BP have been deeply investigated and a convergent fixed-point iterative scheme for its realization has been proposed in [80], and some are presented in section 3.1. The constant γ is set equal to 1, following [85], which gives the following rule for the regularization parameter selection:

$$\lambda = \frac{\|\boldsymbol{y} - \boldsymbol{K}_e\boldsymbol{x}_1 - \mathcal{F}_2(\boldsymbol{x}_2)\|_2^2 + \eta\|\boldsymbol{x}_1\|_2^2}{\|\boldsymbol{x}_1\|_1}$$

6.2.3 The parameter identification method

The proposed iterative method for the identification of both the NMRD parameters $\boldsymbol{f}$, $\boldsymbol{\psi}$ and R_0 and the regularization parameter λ is outlined in Algorithm 10 where, given an initial guess for λ, at each iteration, the NMRD parameters are computed by solving problem (6.10) by the GS method and the regularization parameter value is updated by the BP until the following convergence condition is met:

$$|\lambda^{(k+1)} - \lambda^{(k)}| \leq \mathrm{Tol}_\lambda|\lambda^{(k)}| \quad , \qquad \mathrm{Tol}_\lambda > 0 \tag{6.18}$$

This method is referred to as *AURORA* (Automatic L_1-Regularized Model-Free Analysis).

Algorithm 10 *AURORA*

1: Set $k = 0$, $\eta = 10^{-10}$ and choose a starting guess $\lambda^{(0)}$.
2: **repeat**
3: $k = k + 1$
4: **NMRD and QRE parameters update**

By Algorithm 9 compute $(\boldsymbol{x}_2^{(k)}, \boldsymbol{x}_2^{(k)}) = GS(\boldsymbol{x}_1^{(k-1)}, \boldsymbol{x}_2^{(k-1)})$ i.e.

$$(\boldsymbol{x}_1^{(k)}, \boldsymbol{x}_2^{(k)}) \in \underset{\substack{\boldsymbol{x}_1\in X_1\\ \boldsymbol{x}_2\in X_2}}{\arg\min} \|\boldsymbol{y} - \boldsymbol{K}_e\boldsymbol{x}_1 - \mathcal{F}_2(\boldsymbol{x}_2)\|_2^2 + \lambda^{(k)}\|\boldsymbol{x}_1\|_1 + \eta\|\boldsymbol{x}_2\|_2^2$$

5: **Regularization parameter update**

$$\lambda^{(k+1)} = \frac{\|\boldsymbol{y} - \boldsymbol{K}_e\boldsymbol{x}_1^{(k)} - \mathcal{F}_2(\boldsymbol{x}_2^{(k)})\|_2^2 + \eta\|\boldsymbol{x}_2^{(k)}\|_2^2}{\|\boldsymbol{x}_1^{(k)}\|_1}$$

6: **until** `convergence condition` (6.18)
7: **return** $(\boldsymbol{f}, R_0) = \boldsymbol{x}_1^{(k)}$ and $\boldsymbol{\psi} = \boldsymbol{x}_2^{(k)}$ ▷ Result $(\boldsymbol{f}, R_0, \boldsymbol{\psi})$

Following the analysis of the BP performed in [80], algorithm *AURORA* can be viewed as a fixed point-like scheme (Algorithm 4) for the problem

$$(\boldsymbol{x}_1^*, \boldsymbol{x}_2^*) = \underset{\substack{\boldsymbol{x}_1\in X_1\\ \boldsymbol{x}_2\in X_2}}{\arg\min} \|\boldsymbol{y} - \boldsymbol{K}_e\boldsymbol{x}_1 - \mathcal{F}_2(\boldsymbol{x}_2)\|_2^2 + \lambda^*\|\boldsymbol{x}_1\|_1 + \eta\|\boldsymbol{x}_1\|_2^2$$

$$\lambda^* = \frac{\|\boldsymbol{y} - \boldsymbol{K}_e \boldsymbol{x}_1^* - \mathcal{F}_2(\boldsymbol{x}_2^*)\|_2^2 + \eta \|\boldsymbol{x}_1^*\|_2^2}{\|\boldsymbol{x}_1^*\|_1}$$

The monotone convergence of the sequence $\{\lambda^{(k)}\}$ generated by the fixed point scheme has been proved in [80] when $\lambda^{(0)}$ is chosen in an interval containing only one solution of equation (6.17).

6.3 Results and Discussion

In this section, the results obtained from a set of numerical experiments are presented and discussed to evaluate the proposed algorithm's accuracy, robustness, and efficiency.

First, the experimental setting is described in 6.3.1. The Subsection 6.3.2 presents an evaluation of the algorithm, denoted as *AURORA*, on a synthetic NMRD profile R_1, which is computed using the model (6.1) with predefined values for the parameters $\boldsymbol{\psi}$, $\boldsymbol{f}$, and R_0. The computational efficiency and accuracy of *AURORA* are assessed by comparison with several algorithms available in the MATLAB Optimization Toolbox. Additionally, the robustness of the algorithm is examined in the presence of data noise.

Finally, Subsection 6.3.3 reports the results of the analysis of NMRD profiles obtained from two distinct samples: Dry Nanosponge (DN) and Parmigiano-Reggiano (PR) cheese.

6.3.1 Numerical Experimental Setting

Numerical computations were carried out using Matlab R2022b on a laptop equipped with an Apple M1 processor with 16 GB of 2133 MHz RAM.

It should be noted that throughout the section, the frequencies ν are used instead of the angular frequencies ω, where $\nu \equiv \omega/(2\pi)$.

For all tests, the values $\bar{C}$ and $\bar{\tau}$ in the constraints set $\mathcal{B}_\psi$ (6.9) are set equal to a value large enough so that the intermediate solutions $\psi_1^{(k)}$ and $\psi_4^{(k)}$ never reach such bounds. The suitable values for the test chosen have been $\bar{C} = \bar{\tau} = 100$.

Moreover, the interval $[\omega_\ell, \omega_u]$ in (6.9), representing the region where R_1 interrupts its decaying behaviour due to QRE, is defined by inspection of the NMRD profile. The starting guess for the parameter $\psi_1^{(0)} \equiv C^{HN}$ is obtained by the literature [4]:

$$C^{HN} = \frac{2}{3}\left(\frac{\mu_0}{4\pi}\frac{\gamma_H \gamma_N \hbar}{r_{HN}^3}\right)^2 \approx 0.18 \ \left[\frac{\mu s}{s^2}\right] \tag{6.19}$$

where the values of the physical constants are reported in Table 6.1.

Table 6.1: Characteristic constants for C^{HN} in (6.19).

Constant	Description	Value
μ_0	permeability of vacuum	10^{-7} $[T^2 J^{-1} m^3]$
γ_H	^1H gyromagnetic factor	$2.577\ 10^6$ $[T^{-1} s^{-1}]$
γ_N	^{14}N gyromagnetic factor	$3.078\ 10^6$ $[T^{-1} s^{-1}]$
$\hbar$	reduced Planck's constant	$1.05472\ 10^{-34}$ $[J\ s]$
r_{HN}	^1H $-^{14}N$ inter-spin distance	$1.4\ 10^{-10}$ $[m]$

Concerning the quadrupolar parameters, $\psi_2^{(0)} \equiv \sin^2\Theta^{(0)}$, $\psi_3^{(0)} \equiv \sin^2\Phi^{(0)}$, describing the orientation of the ^1H $-^{14}N$ dipole–dipole axis with respect to the principal axis system of the electric field gradient at

the position of ^{14}N, the initial values are equal to the mean of the corresponding upper and lower bounds in $\mathcal{B}_{\boldsymbol{\psi}}$, i.e., 1/2.

The initial value of $\psi_4^{(0)} \equiv \tau_Q$, i.e., the correlation time characterizing fluctuations of the $^1H - ^{14}N$ dipole-dipole coupling, is set to 1, while $\psi_5^{(0)} \equiv \omega_-^{(0)}$, and $\psi_6^{(0)} \equiv \omega_+^{(0)}$ are defined as follows:

$$\psi_5^{(0)} = \omega_\ell + \frac{1}{4}|\omega_u - \omega_\ell| \quad , \qquad \psi_6^{(0)} = \omega_u - \frac{1}{4}|\omega_u - \omega_\ell|$$

they correspond to the peaks of the QRE observed in the NMRD profile.

The computed results are evaluated by the Mean Squared Error (MSE)

$$\text{MSE} = \frac{\|R_1 - \mathcal{F}(\boldsymbol{f}, \boldsymbol{\psi}, R_0)\|^2}{m}$$

and the Parameter Relative Error (PRE):

$$\text{PRE}(x) = \frac{\|x^{exact} - x^{computed}\|^2}{\|x^{exact}\|^2}$$

with x representing either the vector $\boldsymbol{f}$ or the scalars R_0, ψ_i, $i = 1, \ldots, 6$.

The components of the vector $\boldsymbol{\psi}$ are referenced by the name in the physical model (6.3), according to the mapping introduced in section 6.1, and reported in Table (6.2) for convenience.

Table 6.2: Quadrupolar parameters mapping.

C^{HN}	Φ	Θ	τ_Q	ω_-	ω_+
ψ_1	$\text{asin}(\sqrt{\psi_2})$	$\text{asin}(\sqrt{\psi_3})$	ψ_4	ψ_5	ψ_6

All the tests apply Algorithm 10 with $\text{Tol}_\lambda = 10^{-2}$ in (6.18) and Algorithm 9 with $\text{Tol}_{GS} = 10^{-6}$ in (6.13).

The computational cost is evaluated in terms of execution time and number of iterations.

Finally, regarding the test performed on real experimental data, the *dispersion analysis* 5.3.3 has been performed to evaluate the robustness of the algorithm and to extract the confidence intervals of the computed parameters.

6.3.2 Synthetic Test Problem

The algorithm *AURORA* has been initially tested on the synthetic NMRD profile R_1, shown in Figure 6.1(a) to examine the properties.

This profile is obtained by setting the parameters of model (6.1) as specified in the second column of Table 6.3, with the distribution function $\boldsymbol{f}^*$ depicted in red in Figure 6.2(a).

The accuracy of the computed results can be appreciated in the correlation distribution $\boldsymbol{f}$ and R_1 curves shown in Figure 6.2.

The convergence behaviour has been tested by evaluating the PRE and the MSE at each step of the GS method in the algorithm 9. Figure 6.3 (a) shows the the behaviour of the relative errors for each parameter $(\boldsymbol{f}, R_0, C^{HN}, \Phi, \Theta, \tau_Q, \nu_-, \nu_+)$ compared to their reference values.

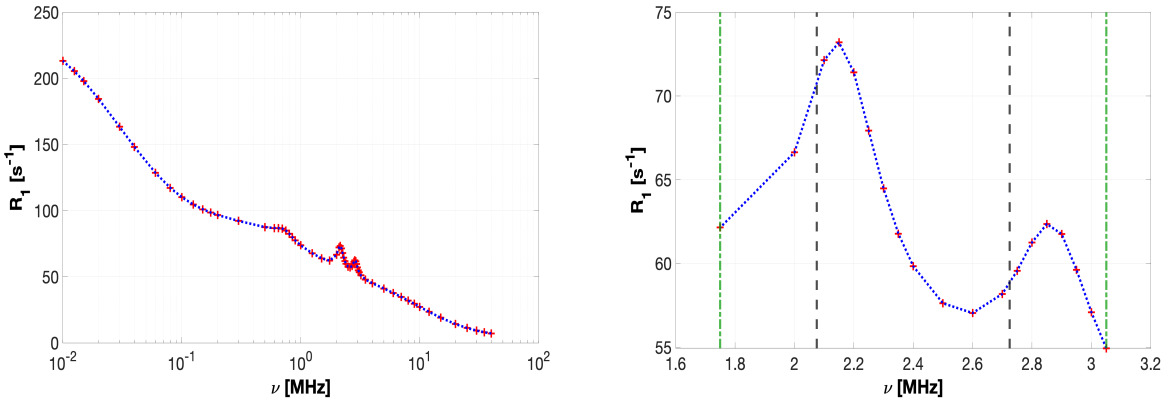
The convergence to reference parameter values is initially non-monotonic for most parameters except τ_Q and ν_-. On the contrary, MSE has a monotonic decrease as reported in Figure 6.3 (b).

The values of the computed parameters and relative errors reported in the third and fourth columns of Table 6.3 confirm the excellent accuracy obtained by the proposed algorithm.

The computed value of the regularization parameter is $\lambda^* = 1.216\ 10^{-9}$ with computation time of $90.44 \pm 0.3\ s$.

Table 6.3: Model parameters: reference (second column), *AURORA* computed values (third column) and PRE (fourth column).

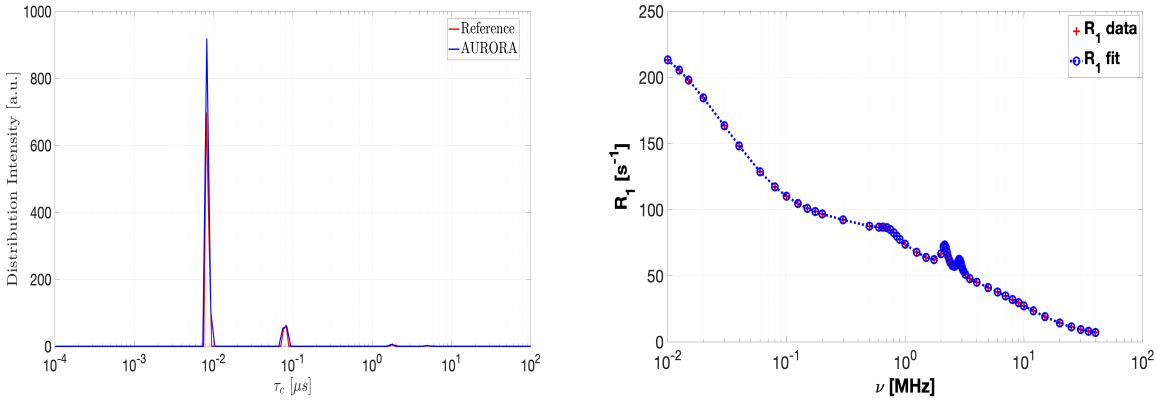
	reference	computed	PRE
R_0	3.69	3.6868	7.0267 10^{-4}
C^{HN}	18.84	18.8453	6.1449 10^{-5}
τ_Q	0.96	0.9554	8.5033 10^{-6}
Θ	1.09	1.0901	6.1449 10^{-5}
Φ	0.57	0.5696	6.9199 10^{-4}
ν_-	2.15	2.1502	5.7363 10^{-6}
ν_+	2.87	2.8696	1.1316 10^{-6}



(a) Full profile.

(b) Zoom of the profile.

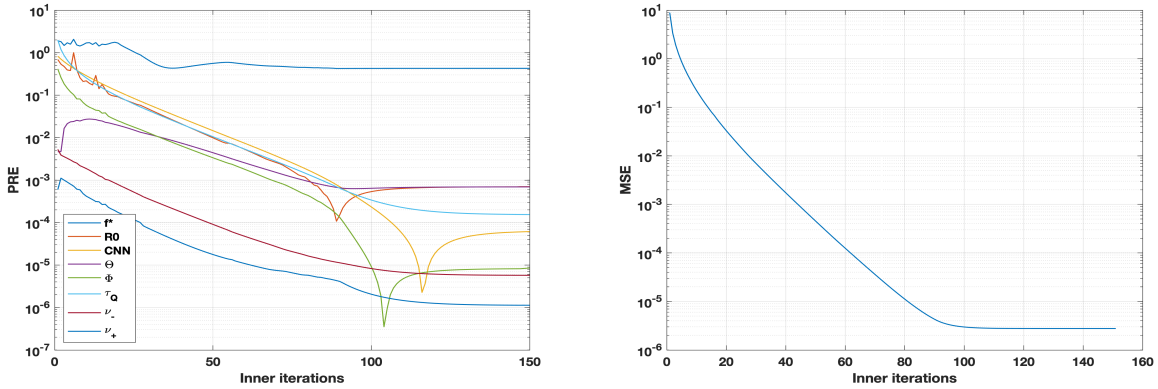
Figure 6.1: Synthetic sample NMRD profile. In Figure (b) the zoom is shown in the reference interval $[\nu_\ell, \nu_u]$ represented by the left and right green vertical lines. Left and right black vertical lines represent the values $\frac{\psi_5^{(0)}}{(2\pi)}$, $\frac{\psi_6^{(0)}}{(2\pi)}$ respectively.



(a) Correlation time distribution.

(b) NMRD profile.

Figure 6.2: Synthetic sample computed results (in blue) obtained by the AURORA algorithm compared to reference one (in red).



(a) PRE values per iteration.

(b) MSE values per iteration.

Figure 6.3: Synthetic R_1 results to test the convergence of the algorithm.

The computation cost, evaluated in terms of inner iterations of the two Gauss-Seidel blocks in Algorithm 9, consists of 147 iterations of the Newton Projection method and 74258 iterations of the truncated Newton interior point method. Note that the greater computational weight lies in truncated Newton interior-point iterations due to the larger size of problem (6.15) compared to (6.16).

Although the convergence of the update formula (6.17) depends on the initial guess $\lambda^{(0)}$, the convergence for $\lambda^{(0)}$ has been found in a quite large interval ($[10^{-16}, 10^{0}]$). In Figure 6.4, the sequences $\lambda^{(k)}$, $k = 0, \ldots, 15$ obtained by Algorithm 10 with $\lambda^{(0)} \in \{10^{-16}, 10^{-6}, 10^{-4}, 10^{-2}, 10^{0}\}$ are represented. Optimal convergence ($k = 1$) is obtained for $10^{-16} \leq \lambda^{(0)} \leq 10^{-4}$ while $\lambda^{(0)} > 10^{-4}$ causes a slight increase of the iterations number, still preserving the convergence up to $\lambda^{(0)} = 1$, which is usually considered as a standard starting guess. Therefore, to keep computations efficient, $\lambda^{(0)} = 10^{-6}$ is used throughout the numerical experiments of this section.

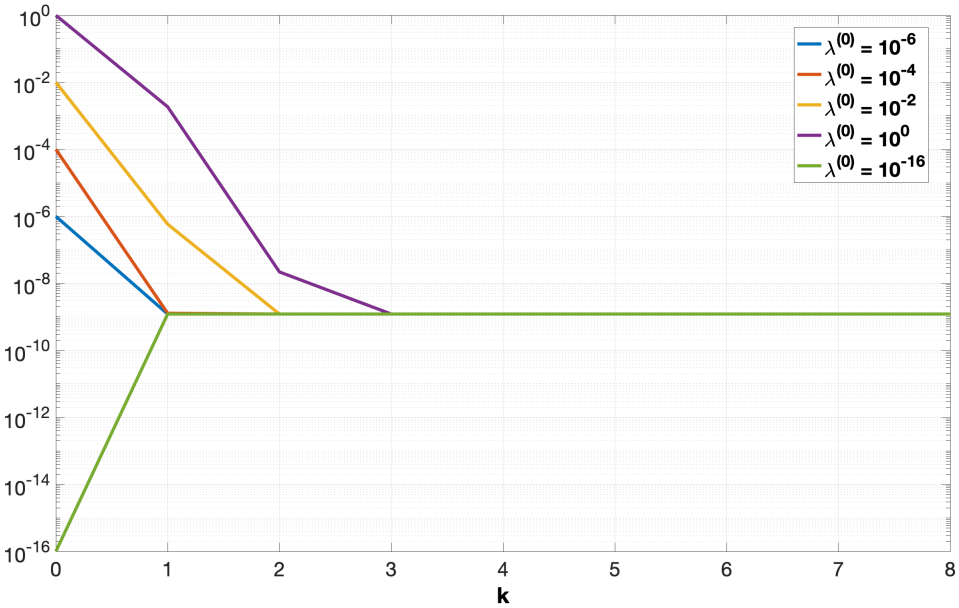


Figure 6.4: Synthetic R_1. Sequence $\{\lambda^{(k)}\}$, obtained by *AURORA* with $\lambda^{(0)} \in \{10^{-16}, 10^{-6}, 10^{-4}, 10^{-2}, 10^{0}\}$.

Comparison with Matlab solvers

This test problem is designed to compare *AURORA* with several methods implemented by the MATLAB function `fmincon`, including the interior-point (`ip`), active-set (`as`), sequential quadratic programming (`sqp`), and trust-region-reflective (`trr`) methods.

It is important to note that *AURORA* automatically computes the value of the regularization parameter λ, whereas the MATLAB function `fmincon` solves the optimization problem (6.10) for a fixed value of λ. Consequently, the GS Algorithm 9 is compared with `ip`, `as`, `sqp`, and `trr` using the same fixed value $\lambda = 1 \cdot 10^{-8}$, which was heuristically determined to be suitable for all methods.

Besides the automatic computation of the regularization parameter λ, *AURORA* splits the unknown parameters into two blocks and alternatively minimizes the objective function for $(R_0, \boldsymbol{f})$, the offset and correlation distribution, and for the quadrupolar parameters $\boldsymbol{\psi}$. Two different methods are used for the solution of the corresponding sub-problems. On the contrary, `fmincon` computes all the parameters applying the same method.

Table 6.4 shows the PRE and MSE values (last row) obtained by *AURORA* (second column) and by the Matlab solvers, highlighting the smallest values.

Table 6.4: Parameter relative errors (**PRE**) and **MSE** of *AURORA* and methods implemented by the Matlab function `fmincon`.

	PRE				
Parameter	*AURORA*	`ip`	`active-set`	`sqp`	`trr`
$\boldsymbol{f}$	$\mathbf{4.2834 \cdot 10^{-1}}$	1.5509	1.4497	1.3020	$8.5279 \cdot 10^{-1}$
R_0	$\mathbf{7.0032 \cdot 10^{-4}}$	$9.9629 \cdot 10^{-1}$	1.0000	2.7930 10^{-1}	$1.3671 \cdot 10^{-1}$
C^{HN}	$5.8238 \cdot 10^{-5}$	4.2908	$9.6353 \cdot 10^{-1}$	$\mathbf{1.5045 \cdot 10^{-5}}$	1.1591 10^{-2}
Θ	$\mathbf{6.9108 \cdot 10^{-4}}$	$6.5929 \cdot 10^{-2}$	$7.7862 \cdot 10^{-2}$	7.2072 10^{-4}	$1.5758 \cdot 10^{-2}$
Φ	$\mathbf{8.7093 \cdot 10^{-6}}$	$5.5535 \cdot 10^{-1}$	2.1372	2.6548 10^{-5}	$7.2619 \cdot 10^{-3}$
τ_Q	$\mathbf{1.5660 \cdot 10^{-4}}$	$9.9228 \cdot 10^{-1}$	$3.1584 \cdot 10^{1}$	1.8903 10^{-4}	$1.1033 \cdot 10^{-2}$
ν_-	$5.7679 \cdot 10^{-6}$	$4.0856 \cdot 10^{-1}$	$2.2756 \cdot 10^{-1}$	$\mathbf{5.6228 \cdot 10^{-6}}$	$5.9438 \cdot 10^{-5}$
ν_+	$\mathbf{1.1391 \cdot 10^{-6}}$	$5.5197 \cdot 10^{-2}$	$3.3362 \cdot 10^{-2}$	$1.2084 \cdot 10^{-6}$	$1.8516 \cdot 10^{-5}$
MSE	$\mathbf{2.8131 \cdot 10^{-6}}$	9.1906	9.0766	$3.1658 \cdot 10^{-6}$	$2.8289 \cdot 10^{-3}$

The distribution $\boldsymbol{f}$ computed by `sqp` is shown in Figure 6.5.

The algorithm *AURORA* has globally superior accuracy both in data fitting, and parameter estimation. Only `sqp` reaches an MSE value ($3.1658 \cdot 10^{-6}$) similar to *AURORA* ($2.8131 \cdot 10^{-6}$), and a slightly better PRE for parameters C^{HN} and ν_-, but the amplitude distribution in Figure 6.5 shows too many spurious peaks. Regarding all the other methods, the obtained results are much larger values compared to *AURORA*.

Test with Noisy Data

In this paragraph, the robustness of the algorithm to data perturbations is assessed by generating noisy data $\boldsymbol{y}^\delta \in \mathbb{R}^m$ from a random uniformly distributed vector $\boldsymbol{v} \in \mathbb{R}^m$ with values in the interval $[-1, 1]$, such that

$$y_i^\delta = y_i(1 + \delta v_i) \quad , \qquad i = 1, \ldots, m.$$

The cases $\delta = 1\%, 5\%, 10\%$ are considered.

The computation times (averaged over 10 runs) and the iteration numbers, presented in Table 6.5, indicate that both the iteration numbers and computation times decrease as the noise percentage increases. This

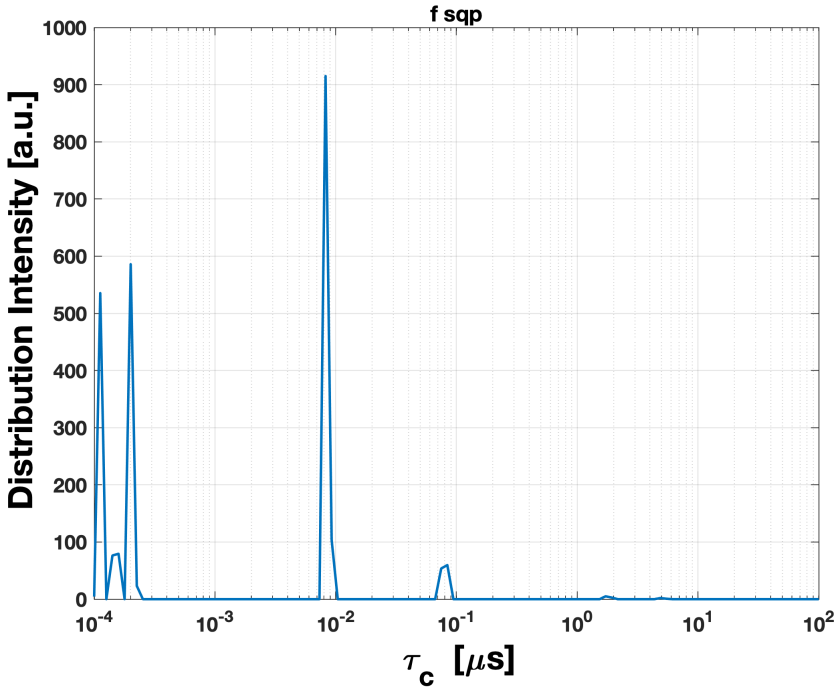


Figure 6.5: Correlation time distribution $\boldsymbol{f}$ computed by `sqp` method.

behaviour can be attributed to the effect of increasing noise on the computed regularization parameters, as shown in the third column of Table 6.5.

Table 6.5: Computation times and iteration numbers with noise $\delta = 1\%, 5\%10\%$.

δ	Time	Total iterations (6.15)	Total iterations (6.16)	λ
$1 \cdot 10^{-2}$	38.39 ± 0.28	52217	89	$5.8405\ 10^{-5}$
$5 \cdot 10^{-2}$	14.39 ± 0.16	44414	72	$1.8803\ 10^{-3}$
$1 \cdot 10^{-1}$	4.90 ± 0.10	16846	28	$9.3561\ 10^{-3}$

The relations between the execution times are not found precisely in the number of iterations because they do not take into account the backtracking steps internal to the Newton method.

By computing 500 noisy samples $\boldsymbol{y}_j^\delta$, the *AURORA* algorithm is executed, and the errors on the estimated parameters as well as the reconstructed NMRD profiles are compared.

For the noise levels $\delta = 1\%, 5\%, 10\%$, the mean Parameter Relative Error for each parameter is calculated. The mean values are represented in the bar plot shown in Figure 6.6, alongside the product $C^{HN} \cdot \tau_Q$.

The mean PRE and MSE are reported in Table 6.6.

The computed R_1 curves and the zoom in the QRE interval are shown in Figures 6.7, 6.8, and 6.9 for $\delta = 1\%, 5\%, 10\%$ respectively.

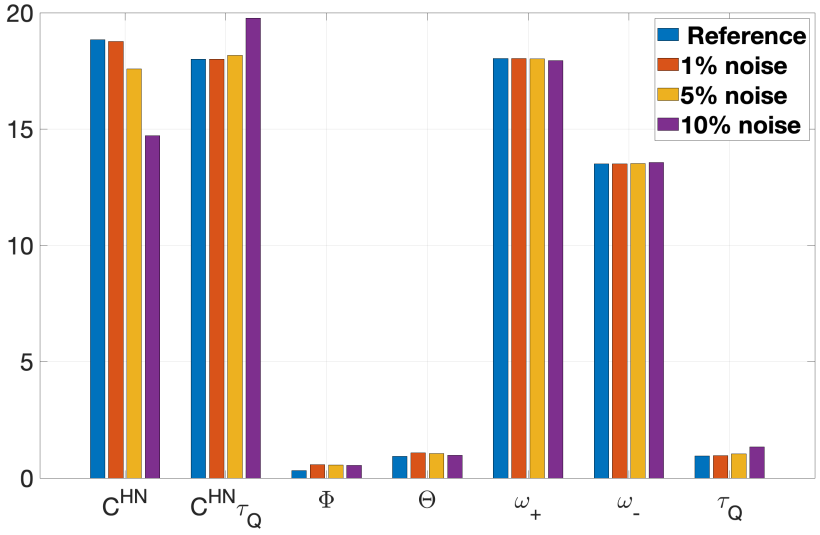
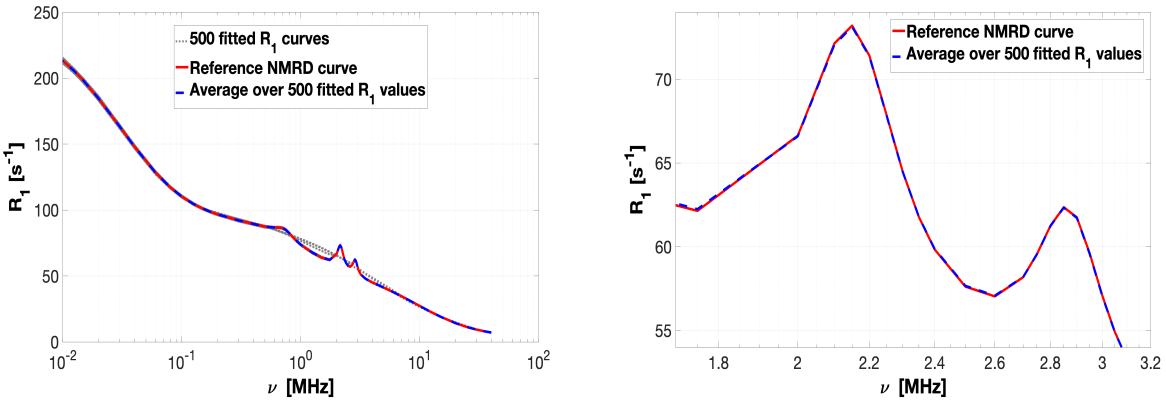


Figure 6.6: Mean parameter values computed by 500 noisy NMRD profiles with noise $\delta = 1\%, 5\%10\%$.

Table 6.6: Mean **PRE** and **MSE** on 500 noisy NMRD profiles with $\delta = 1\%, 5\%, 10\%$.

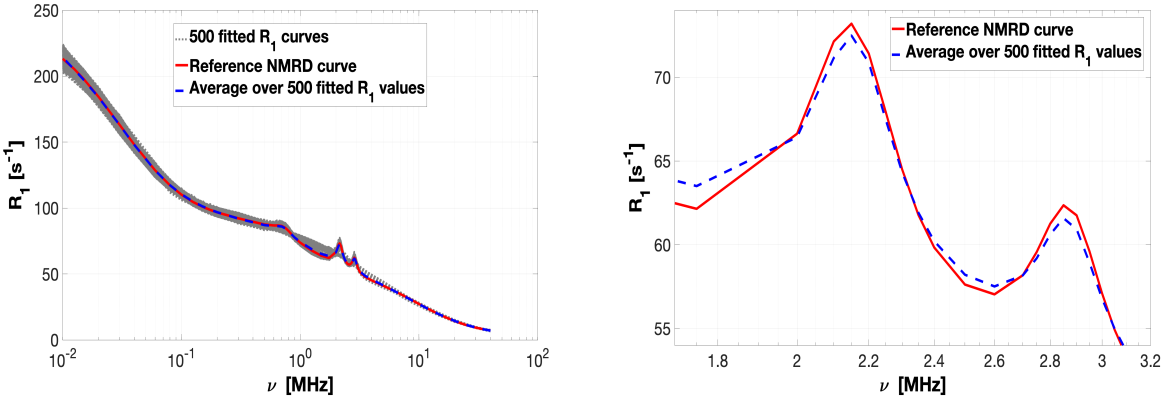
	PRE		
	1%	5%	10%
$\boldsymbol{f}$	$5.9019 \cdot 10^{-1}$	1.1816	1.4509
R_0	$3.6393 \cdot 10^{-2}$	$1.6726 \cdot 10^{-1}$	$1.8099 \cdot 10^{-1}$
C^{HN}	$3.3625 \cdot 10^{-2}$	$2.7021 \cdot 10^{-1}$	$4.7742 \cdot 10^{-1}$
Θ	$2.3023 \cdot 10^{-2}$	$1.0678 \cdot 10^{-1}$	$2.1726 \cdot 10^{-1}$
Φ	$3.5151 \cdot 10^{-2}$	$4.0280 \cdot 10^{-1}$	$6.5910 \cdot 10^{-1}$
τ_Q	$4.4998 \cdot 10^{-2}$	1.8862	$1.1095 \cdot 10^{1}$
ν_-	$4.3917 \cdot 10^{-3}$	$4.8712\ 10^{-2}$	$7.2441 \cdot 10^{-2}$
ν_+	$3.0889 \cdot 10^{-3}$	$3.8712\ 10^{-2}$	$5.6856 \cdot 10^{-2}$
MSE	$1.5980 \cdot 10^{-1}$	3.1441	$1.0055 \cdot 10^{1}$



(a) Full profile.

(b) Zoom of the profile in QRE interval.

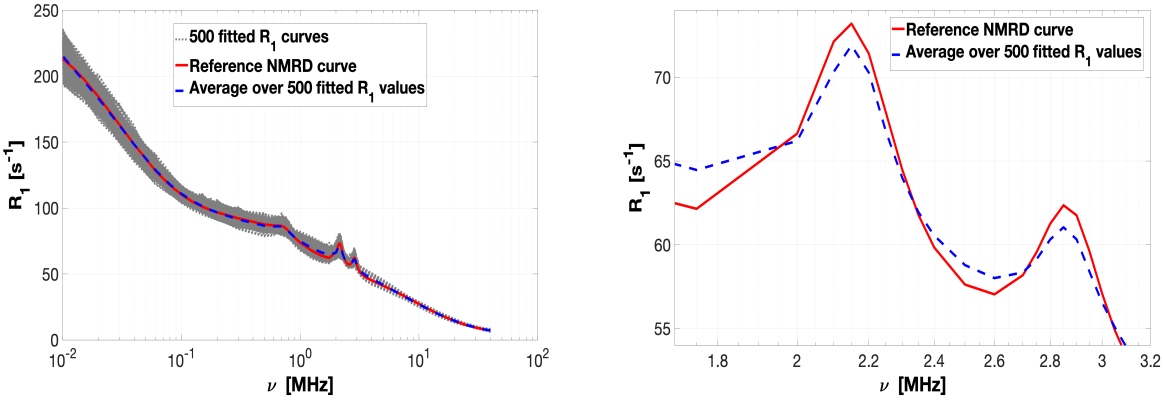
Figure 6.7: Fit of NMRD obtained from 500 noisy Synthetic NMRD curves with noise $\delta = 1\%$, where the 500 fitted R_1 curves are in light grey, the reference NMRD curve is in red, and the average over 500 fitted R_1 values is in blue.



(a) Full profile.

(b) Zoom of the profile in QRE interval.

Figure 6.8: Fit of NMRD obtained from 500 noisy Synthetic NMRD curves with noise $\delta = 5\%$, where the 500 fitted R_1 curves are in light grey, the reference NMRD curve is in red, and the average over 500 fitted R_1 values is in blue.



(a) Full profile.

(b) Zoom of the profile in QRE interval.

Figure 6.9: Fit of NMRD obtained from 500 noisy Synthetic NMRD curves with noise $\delta = 10\%$, where the 500 fitted R_1 curves are in light grey, the reference NMRD curve is in red, and the average over 500 fitted R_1 values is in blue.

In Figure 6.6, it is observed that data noise primarily affects the values of C^{HN}, τ_Q, and Φ. However, when considering the product $C^{HN}\tau_Q$, represented by the second group in Figure 6.6, it is evident that the value is preserved for noise levels $\delta = 1\%$ and 5%. This behavior reflects a physical characteristic, supporting the accuracy of the associated parameters.

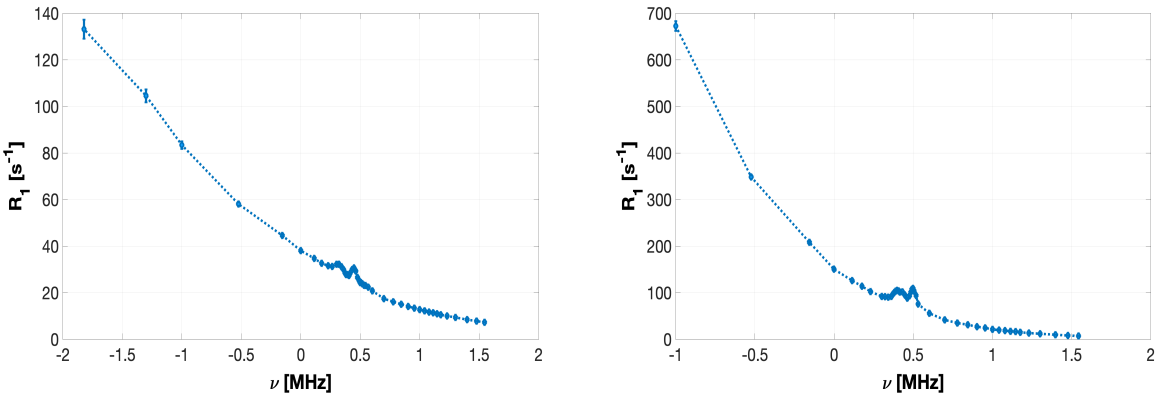
Although the average MSE increases with data noise, the computed average R_1 curves show a very good agreement with the reference NMRD profiles (Figures 6.7, 6.8 and 6.9). The QRE is well reproduced even with high noise (Figures 6.7 (b), 6.8 (b) and 6.9 (b)).

6.3.3 NMRD Profiles from FFC Measures

The NMRD profiles obtained from two different materials described in [26] have been considered and tested.

- **A sample of 24-month-aged Parmigiano-Reggiano (PR) cheese.** The NMRD profile represented in Figure 6.10 (a) has $m = 48$ values with confidence intervals ranging from $\pm 0.35\%$ to $\pm 3.07\%$ of the value. The quadrupolar peaks, represented in Figure 6.11 (a), correspond to frequency values $\nu_- = 2.1$ and $\nu_+ = 2.8$ of values $R1_- = 32.2\ s^{-1}$ and $R1_+ = 30.7\ s^{-1}$ respectively.

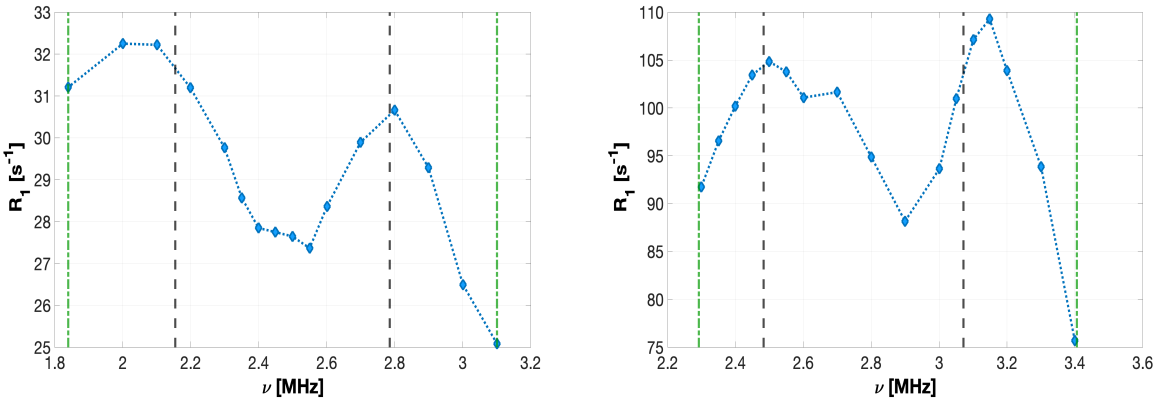
- **A sample of Dry nanosponge (DN).** In this case the NMRD profile represented in Figure 6.10 (b) has $m = 44$ values with confidence intervals ranging from $\pm 0.47\%$ to $\pm 1.54\%$ of the value. The quadrupolar peaks, represented in Figure 6.10 (b), correspond to frequency values $\nu_- = 2.4991\ MHz$ and $\nu_+ = 3.1488\ MHz$ of values $R1_- = 104.85\ s^{-1}$ and $R1_+ = 104.85\ s^{-1}$ respectively.



(a) Parmigiano-Reggiano (PR) sample.

(b) Dry Nanosponge (DN) sample.

Figure 6.10: NMRD profiles.



(a) Parmigiano-Reggiano (PR) sample.

(b) Dry Nanosponge (DN) sample.

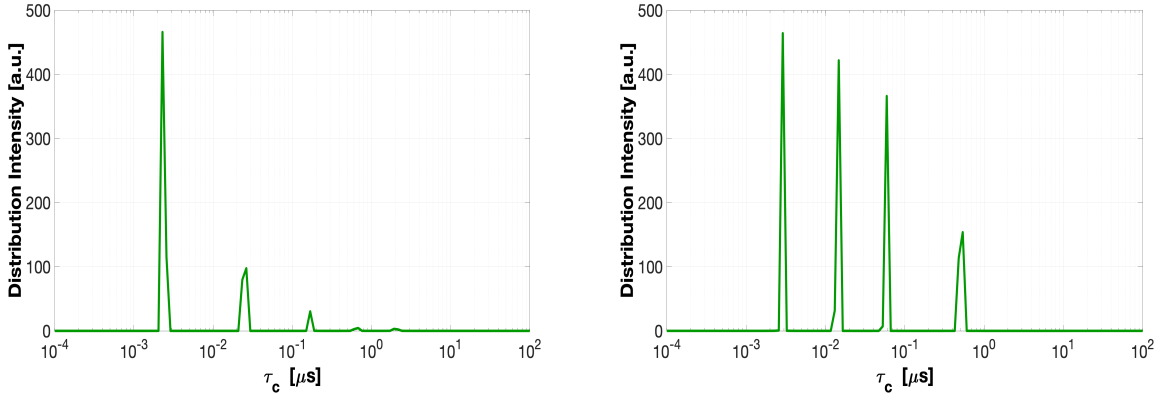
Figure 6.11: Zoom of the profile in QRE interval.

The proposed *AURORA* method has been used to compute the model parameters reported in Table 6.7. The obtained correlation distributions are represented in Figure 6.12 in a dark green line.

Concerning the fit of the NMRD profiles, the MSE has been measured and reported in the last row of Table 6.7. Notably, the PR sample exhibits a low MSE of $7.8887 \cdot 10^{-2}$, reflecting a high degree of accuracy in capturing the underlying data behavior. In contrast, the DN sample yields an MSE of 2.7853, indicating a

Table 6.7: Values of the parameters fitted by *AURORA* and MSE in the last row.

	Parameter values	
	PR	DN
R_0	3.23	2.73
C^{NH}	5.66	69.00
Θ	1.25	0.91
Φ	0.86	0.87
τ_Q	1.02	0.74
ν_-	2.1	2.56
ν_+	2.8	3.17
MSE	$7.8887 \cdot 10^{-2}$	2.7853



(a) Parmigiano-Reggiano (PR) sample.

(b) Dry Nanosponge (DN) sample.

Figure 6.12: Correlation distribution (dark green lines).

less precise identification. This discrepancy likely arises from intrinsic differences in the data characteristics of the two samples.

The fitted NMRD profiles, represented in Figure 6.13, show in blue line the data and error bars while the fitted curves are represented in red line for both samples.

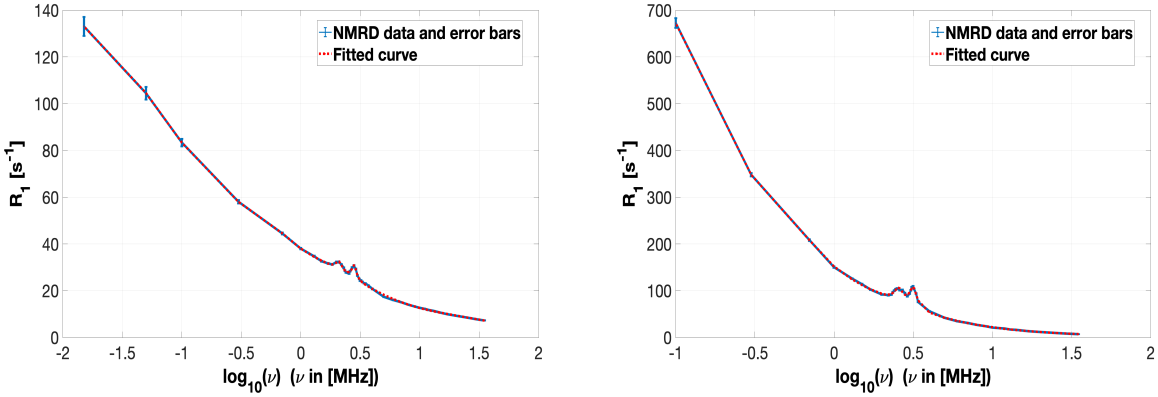
The zoom in the frequencies of the QRE interval is shown in Figure 6.14.
The results confirm the excellent fit to the NMRD profile (Figure 6.13) also in the QRE interval (Figure 6.14).

Dispersion Analysis on Profiles with QRE

Finally, the dispersion analysis introduced in the previous chapter, in section 5.3.3, has been applied in this case of the *AURORA* algorithm analysing NMRD curves with QRE effect to extract confidence intervals of the computed parameters.

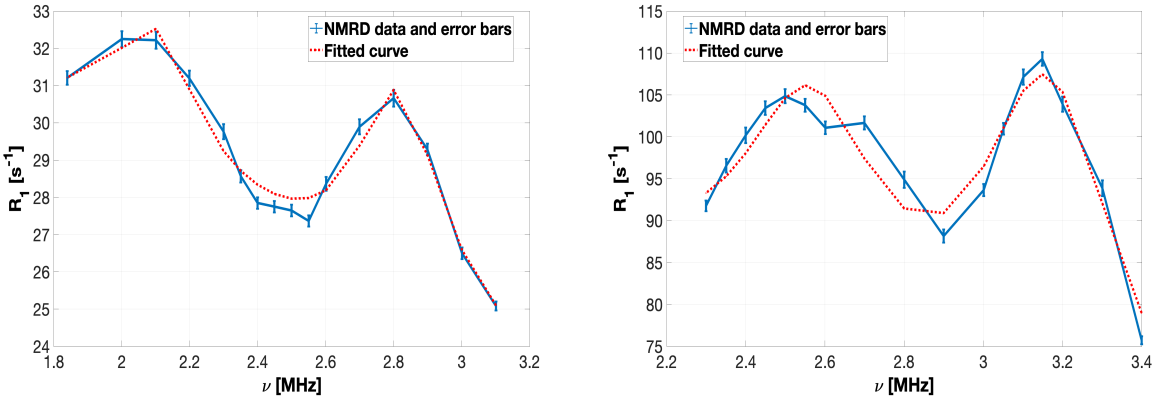
Tables 6.8, and 6.9 compare the computed parameters, i.e., $\{R_0, C^{HN}, \Theta, \Phi, \tau_Q, \nu_-, \nu_+\}$, confidence intervals [164], mean, and medians for both Parmigiano and Nanosponge samples obtained by the algorithm.

For the PR sample, parameters such as R_0, C^{HN}, and τ_Q exhibit relatively narrow confidence intervals and small discrepancies between the mean and median, indicating consistent and symmetric distributions. In contrast, the DN sample demonstrates broader confidence intervals, particularly for C^{HN}, suggesting



(a) Parmigiano-Reggiano (PR) sample. (b) Dry Nanosponge (DN) sample.

Figure 6.13: NMRD data and error bars (blue lines) and fitted curve (red lines).



(a) Parmigiano-Reggiano (PR) sample. (b) Dry Nanosponge (DN) sample.

Figure 6.14: Zoom of data and fitted curves in the QRE intervals. NMRD data and error bars (blue lines) and fitted curves (red lines).

Table 6.8: Comparison of the Confidence Intervals, Mean, and Median for PR sample.

	Confidence Interval	Mean	Median
R_0 $[s^{-1}]$	[2.316, 4.013]	$3.308{\cdot}10^0$	$3.252{\cdot}10^0$
C^{HN} $[\mu s/s^2]$	[5.285, 6.351]	$5.736{\cdot}10^0$	$5.733{\cdot}10^0$
Θ[rad]	[1.058, 1.253]	$1.214{\cdot}10^0$	$1.253{\cdot}10^0$
Φ[rad]	[0.828, 0.891]	$0.860{\cdot}10^0$	$0.859{\cdot}10^0$
τ_Q $[\mu s]$	[0.937, 1.106]	$1.020{\cdot}10^0$	$1.020{\cdot}10^0$
ν_- [MHz]	[2.080, 2.111]	$2.096{\cdot}10^0$	$2.095{\cdot}10^0$
ν_+ [MHz]	[2.808, 2.821]	$2.815{\cdot}10^0$	$2.814{\cdot}10^0$

greater variability in this computed parameter. Across both samples, the parameters ν_- and ν_+ show negligible differences between the mean and median, reflecting symmetric distributions. These findings underscore differences in variability and central tendencies between the PR and DN samples for the analysed

Table 6.9: Comparison of the Confidence Intervals, Mean, and Median for DN sample.

	Confidence Interval	Mean	Median
R_0 $[s^{-1}]$	[1.905, 3.510]	$2.736 \cdot 10^0$	$2.749 \cdot 10^0$
C^{HN} $[\mu s/s^2]$	[57.865, 70.670]	$6.011 \cdot 10^1$	$6.010 \cdot 10^1$
Θ[rad]	[0.891, 1.075]	$1.057 \cdot 10^0$	$1.058 \cdot 10^0$
Φ[rad]	[0.862, 0.883]	$0.872 \cdot 10^0$	$0.873 \cdot 10^0$
τ_Q $[\mu s]$	[0.701, 0.756]	$0.731 \cdot 10^0$	$0.732 \cdot 10^0$
ν_- [MHz]	[2.125, 2.720]	$2.525 \cdot 10^0$	$2.525 \cdot 10^0$
ν_+ [MHz]	[2.989, 3.340]	$3.131 \cdot 10^0$	$3.131 \cdot 10^0$

parameters.

In Appendix A, the obtained results are shown in the scatter plots for the 500 repetitions, reporting some figures for both the computed profiles and the correlation time distributions as examples.

Chapter 7

Neural Network-Based Inversion of NMR Dispersion Profiles

This chapter is based on the publication [167].

Despite the widespread recognition and application of artificial intelligence methods across various disciplines, e.g., magnetic resonance imaging, their utilization in NMR remains relatively under-explored and the adoption of these advanced computational tools is limited. Some AI-based approaches applied to NMR can be found in the literature. For instance, Weisman et al. developed a supervised learning method applied to the food industry. Specifically, a convolutional neural network (CNN) has been trained to classify the transversal relaxation time to identify the oil oxidation level [23]. In NMR spectroscopy, Hansen reconstructs the sparsely sampled spectra acquired by fast acquisition multi-dimensional NMR spectra techniques employing a deep neural network [24].

However, the widespread adoption of FFC-NMR is hindered by the complexity of the analytical instruments and the need for deep expertise in NMR and materials physics. Addressing this challenge, this chapter describes a machine learning framework tailored to the QRE phenomenon, utilizing a neural network to compute the parameters characterizing the quadrupolar relaxation model. A pre-trained feed-forward neural network is embedded into the NMRD data fitting procedure according to the philosophy of the Plug-and-Play (PnP) approach, facilitating integration with NMRD data inversion software based on the coordinate descent method (two-block non-linear Gauss-Seidel) [155, 168].

Since their introduction, PnP methods have become standard tools for computational imaging [134, 169], providing practical approaches to integrating learned models with imaging physics to solve inverse imaging problems. In this work, a similar procedure is followed, embedding the neural network in an alternate minimization procedure.

The contributions of this work can be summarized as follows:

- Design of a neural network to predict QRE parameters.
- Definition of an appropriate training procedure.
- Proposal and application of a novel specific loss function for network training and prediction.

To my knowledge, no applications of PnP methods to NMRD data inversion exist, making the proposed method the first example of applying PnP to NMR-FFC data treatment.

This contribution demonstrates the effectiveness of using neural networks for NMRD profile inversion. The results are tested and compared with those obtained using a previously developed robust optimization

method (Chapter 6, [155]), showing strong concordance. This suggests that neural network-based approaches have the potential to significantly expedite the analysis of extensive datasets in this field.

Following this introduction, section 7.1 details the physical model and the mathematical parameter estimation problem. Section 7.2 describes the proposed neural network approach and the computational framework of the algorithm. Section 7.3 presents and discusses the numerical results from testing on two sets of NMRD profiles, each illustrating significant potential scenarios in food analysis [16].

7.1 Towards an Enhanced Framework

As described in the previous chapters, the relaxation rates can be represented as linear combinations of spectral density functions of the motion modulating the interactions.

Let us briefly recall for convenience the discrete model (6.5) required to describe the last proposed method to extract the parameters and analyze the NMRD curves:

$$\boldsymbol{y} = R_0 + \boldsymbol{K}\boldsymbol{f} + \mathcal{F}_2(\boldsymbol{\psi}) \tag{7.1}$$

where the second term in the right-hand side of (7.1) is the linear function derived from the discretization of the integral for R^{HH} in (6.2) with $\boldsymbol{K} \in \mathbb{R}^{m\times n}$:

$$\boldsymbol{K}_{i,j} = \frac{\tau_j}{(1+(\omega_i\tau_j)^2)} + \frac{4\tau_j}{(1+4(\omega_i\tau_j)^2)} \quad , \qquad i = 1,\dots,m,\ j = 1,\dots,n$$

and $\boldsymbol{f} \in \mathbb{R}^n$, a sparse vector of the sampled correlation distribution function.

The third term in (7.1) models the QRE term $R^{HN}(\omega_i)$, $i = 1,\dots,m$ in (6.3). It is represented by the function $\mathcal{F}_2 : \mathbb{R}^6 \to \mathbb{R}^m$, depending on the quadrupolar parameters $\boldsymbol{\psi} \equiv (\psi_1,\psi_2,\psi_3,\psi_4,\psi_5,\psi_6)^T \equiv (C^{HN}, sin^2(\Theta), sin^2(\Phi), \tau_Q, \omega_-, \omega_+)^T$. Let us rewrite it in the following more compact form:

$$(\mathcal{F}_2(\boldsymbol{\psi}))_i = \psi_1 \begin{bmatrix} \frac{1}{3}+\psi_2(1-\psi_3) \\ \frac{1}{3}+\psi_2\psi_3 \\ \frac{1}{3}+(1-\psi_2) \end{bmatrix}^T \cdot \begin{bmatrix} \frac{\psi_4}{1+(\omega_i-\psi_5)^2\psi_4^2} + \frac{\psi_4}{1+(\omega_i+\psi_5)^2\psi_4^2} \\ \frac{\psi_4}{1+(\omega_i-\psi_6)^2\psi_4^2} + \frac{\psi_4}{1+(\omega_i+\psi_6)^2\psi_4^2} \\ \frac{\psi_4}{1+(\omega_i-\Delta\psi_{5,6})^2\psi_4^2} + \frac{\psi_4}{1+(\omega_i+\Delta\psi_{5,6})^2\psi_4^2)} \end{bmatrix} \tag{7.2}$$

where $\Delta\psi_{5,6} = (\psi_6 - \psi_5)$.

Since the problem is ill-conditioned, and the proposed strategy is an integrated method that employs a machine learning approach, let us also recall the L_1-regularized least squares inverse problem:

$$\begin{aligned} \min_{\boldsymbol{f},\boldsymbol{\psi},R_0} \quad & \|(\boldsymbol{K}\boldsymbol{f} + \mathcal{F}_2(\boldsymbol{\psi}) + R_0) - \boldsymbol{y}\|_2^2 + \lambda\|\boldsymbol{f}\|_1 \\ \text{s.t.} \quad & \boldsymbol{f} \geq \boldsymbol{0} \\ & \boldsymbol{\psi} \in \mathcal{B}_{\boldsymbol{\psi}} \\ & R_0 \geq 0 \end{aligned} \tag{7.3}$$

where the set $\mathcal{B}_{\boldsymbol{\psi}}$ defines the box constraints on $\boldsymbol{\psi}$:

$$\mathcal{B}_{\boldsymbol{\psi}} = \left\{\boldsymbol{\psi} \,:\, \psi_1 \in [0,\bar{C}];\ \psi_2,\psi_3 \in [0,1];\ \psi_4 \in [0,\bar{\tau}];\ \psi_5,\psi_6 \in [\omega_\ell,\omega_u]\right\} \tag{7.4}$$

with the values $\bar{C}$, $\bar{\tau}$, and ω_ℓ, ω_u quite delicate to choose as they depend on the physical properties of the analysed sample and significantly influence the quality of the result. Therefore, they need to be set appropriately, requiring knowledge of NMR and the physical structure of the studied material.

Considering the objective function (7.3), the first term represents the fidelity term of Chapter 2, Φ, and it imposes R_1 data consistency; while the second component is the regularization term, Ψ, introduced to stabilize the problem exploiting a priori knowledge on the sparsity of $\boldsymbol{f}$. The regularization parameter $\lambda > 0$ weights the contribution of the L_1 regularization term. In the previous chapter, a practical and effective solution has been described, consisting of an automatic rule, based on the balancing principle framework [80] for the choice of the regularization parameter λ. Then, for each value of the regularization parameter, a two-block nonlinear Gauss-Seidel method has been used for the solution of (7.3).

This means that, from an algorithmic point of view, there are two main nested loops: an outer balancing principle loop, using the counter ℓ, tailored to update $\lambda^{(\ell)}$ according to the balancing principle, and an inner Gauss-Seidel loop to compute the current solution $(\boldsymbol{f}, \boldsymbol{\psi}, R_0)$ for each $\lambda^{(\ell)}$. In particular, the two-block nonlinear Gauss-Seidel method finds the current solution, for each choice of $\lambda^{(\ell)}$, by alternately approximating the quadrupolar parameters $\boldsymbol{\psi}$ and the components $(\boldsymbol{f}, R_0)$ as follows:

$$(\boldsymbol{f}^{(k+1)}, R_0^{(k+1)}) \in \underset{\boldsymbol{f}\geq 0, R_0 \geq 0}{\arg\min} \|(\boldsymbol{K}\boldsymbol{f} + R_0) - (\boldsymbol{y} - \mathcal{F}_2(\boldsymbol{\psi}^{(k)}))\|_2^2 + \lambda\|\boldsymbol{f}\|_1 + \eta\|\boldsymbol{f}\|_2^2 \tag{7.5}$$

$$\boldsymbol{\psi}^{(k+1)} \in \underset{\boldsymbol{\psi}\in\mathcal{B}_{\boldsymbol{\psi}}}{\arg\min} \|\mathcal{F}_2(\boldsymbol{\psi}) - (\boldsymbol{y} - (\boldsymbol{K}\boldsymbol{f}^{(k+1)} + R_0^{(k+1)}))\|_2^2 \tag{7.6}$$

with $k \geq 0$ and initial guesses $(\boldsymbol{f}^{(0)}, R_0^{(0)})$ and $\boldsymbol{\psi}^{(0)}$. This reformulation of the GS of the solution presented in the previous chapter is applied for each regularization parameter. For simplicity of notation, the iteration index ℓ of the outer loop has been omitted. Moreover, let us remember that the last L_2-based regularization term $\eta\|\boldsymbol{f}\|_2^2$ in (7.5), where η is a fixed small positive value, has been introduced to guarantee the convergence of the proposed Gauss-Seidel method as proven in the last chapter. This subproblem was efficiently solved with the truncated Newton interior-point method [158].

The second subproblem (7.6) is a bound-constrained nonlinear least squares problem. In the last chapter, the Newton projection method [67, 68] was described as a solution for this subproblem, with the Hessian matrix approximated using the Levenberg-Marquardt method [166], due to the ill-conditioning of the Jacobian of $\mathcal{F}_2$. The main challenges of this problem are related to the ill-conditioning of the Jacobian matrix, which requires an appropriate implementation of the Levenberg-Marquardt method, and the need for proper setting of the box constraints. Therefore, the possibility of using a neural network in this context has been explored.

7.2 Proposed Neural Network and Plug-and-Play Algorithm

Inspired by the PnP prior approach described in section 4.2.2, the design of the PnP algorithm where a neural network replaces the Newton projection method for the solution of (7.6) has been investigated. Just as a denoiser can replace the proximal step related to the prior update in the ADMM algorithm [134], the presented proposal consists of using a suitably trained neural network instead of the Newton projection method to update the quadrupolar parameters.

Let us start the description of the proposed neural network by recalling for convenience the changing of variables introduced in the previous chapter:

$$\boldsymbol{x}_1 \equiv (\boldsymbol{f}, R_0) \in \boldsymbol{X}_1 \quad , \qquad \boldsymbol{x}_2 \equiv \boldsymbol{\psi} \in \boldsymbol{X}_2$$

$$\boldsymbol{X}_1 \equiv \{\boldsymbol{x}_1 \geq 0\} \quad , \qquad \boldsymbol{X}_2 \equiv \mathcal{B}_{\boldsymbol{\psi}}$$

and

$$\boldsymbol{K}_e = [\boldsymbol{K} \;\; \mathbf{1}] \in \mathbb{R}^{m\times(n+1)}$$

and rewriting (7.3) as follows:

$$\boldsymbol{x}_1^{(k+1)} \in \underset{\boldsymbol{x}_1 \in \boldsymbol{X}_1}{\arg\min} \|(\boldsymbol{K}_e \boldsymbol{x}_1 - (\boldsymbol{y} - \mathcal{F}_2(\boldsymbol{x}_2^{(k)}))\|_2^2 + \lambda \|\mathcal{D}_{n,0} \boldsymbol{x}_1\|_1 + \eta \|\mathcal{D}_{n,0} \boldsymbol{x}_1\|_2^2 \tag{7.7}$$

$$\boldsymbol{x}_2^{(k+1)} \in \underset{\boldsymbol{x}_2 \in \boldsymbol{X}_2}{\arg\min} \|\mathcal{F}_2(\boldsymbol{x}_2) - (\boldsymbol{y} - \boldsymbol{K}_e \boldsymbol{x}_1^{(k+1)})\|_2^2 \tag{7.8}$$

where $\mathcal{D}_{n,0} = I - e_{n+1} e_{n+1}^T$, I is the identity matrix of size $n+1$ and e_{n+1} is the last column of I. A key observation for conceptualizing this neural network regards one possible interpretation of the Gauss-Seidel algorithm. A closer look at (7.7)−(7.8) reveals that the basic idea of the Gauss-Seidel method is to alternately approximate the quadrupolar component of the NMRD profile $R^{HN*} := \boldsymbol{y}^* - (\boldsymbol{K}_e \boldsymbol{x}_1^*)$, and the non-quadrupolar component $\boldsymbol{y}^* - \mathcal{F}_2(\boldsymbol{x}_2^*)$, where $(\boldsymbol{x}_1^*, \boldsymbol{x}_2^*)$ are the true parameters generating the noiseless observations $\boldsymbol{y}^*$.

Therefore, a neural network $\nu_\Theta : \mathbb{R}^m \to \mathbb{R}^6$ has been proposed. It is required to be able to *invert* the general quadrupolar function $\mathcal{F}_2$ by solving the following constrained least square problem, derived by (7.8):

$$\nu_\Theta(R^{HN(k+1)}) \in \underset{\boldsymbol{x}_2 \in \mathbb{R}^6}{\arg\min} \|\mathcal{F}_2(\boldsymbol{x}_2) - R^{HN(k+1)}\|_2^2 \tag{7.9}$$
$$\text{s.t. } \boldsymbol{x}_2 \in \boldsymbol{X}_2$$

where $R^{HN(k+1)} := \boldsymbol{y} - (\boldsymbol{K}_e \boldsymbol{x}_1^{(k+1)})$ is the $(k+1)$-th *approximation* of the true quadrupolar component R^{HN*}, estimated by means of solution $\boldsymbol{x}_1^{(k+1)}$ of subproblem (7.7). A schematic representation of the proposed method can be found in figure 7.1.

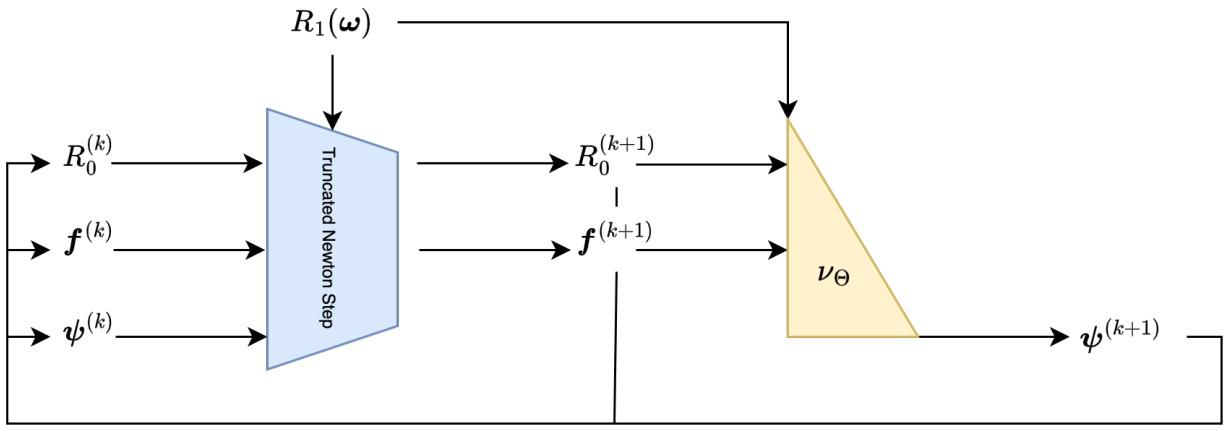


Figure 7.1: PnP-GS scheme.

In summary, this neural network takes as input the current approximation of the quadrupolar component of the NMRD profile $R^{HN(k+1)}$, computed as the difference between the data $\boldsymbol{y}$ and the current approximation of the non-quadrupolar component of the NMRD profile $\boldsymbol{K}_e \boldsymbol{x}_1^{(k+1)}$. The network then outputs the current estimation of the quadrupolar parameters $\boldsymbol{x}_2^{(k+1)} := \nu_\Theta(R^{HN(k+1)})$.

To train ν_Θ for this task, the following procedure has been used: given N sets of quadrupolar parameters $\{\boldsymbol{x}_{2_l}\}_{l=1}^N$, satisfying the physical bounds (7.4), $R_l^{HN} := \mathcal{F}_2(\boldsymbol{x}_{2_l})$ for each $l = 1, \ldots, N$ is computed. Then, the following loss function is defined:

$$\mathcal{L}(\Theta) := \sum_{l=1}^{N} (1-\alpha) \|\nu_\Theta(R_l^{HN}) - \boldsymbol{x}_{2_l}\|_p^p + \alpha \|\mathcal{F}_2(\nu_\Theta(R_l^{HN})) - R_l^{HN}\|_p^p \quad , \qquad p = 1, 2 \tag{7.10}$$

Therefore, the following minimum problem is addressed:

$$\Theta^* = \arg\min_{\Theta} \mathcal{L}(\Theta)$$

The resulting network ν_{Θ^*} computes:

$$\nu_{\Theta^*}(R^{HN(k+1)}) \in \arg\min_{\boldsymbol{x}_2 \in \boldsymbol{X}_2} (1-\alpha)\mathcal{P}(\boldsymbol{x}_2) + \alpha \|\mathcal{F}_2(\boldsymbol{x}_2) - R^{HN(k+1)}\|_2^2 \tag{7.11}$$

where $\mathcal{P}(\boldsymbol{x}_2)$ is a prior term on the quadrupolar parameters learned by the network from the training data, and α is a hyper-parameter to be tuned. This new hyper-parameter can be interpreted as an additional regularization parameter introduced to solve the inverse subproblem (7.9) of inverting $\mathcal{F}_2$. The necessity of including this parameter is discussed in section 7.3.

Concerning the architecture of ν_Θ, a feed-forward fully connected neural network with 7 hidden layers of different dimensions was considered. Specifically, the first one presents 512 neurons, while the next 4 hidden layers have 256 neurons, and the last two have 64 and 32 neurons, respectively. The input layer has a dimension that depends on the length of the acquired experimental signal.

In the presented case, the experimental signals have a length of 45 acquired points therefore input layer has 45 neurons. Finally, the output layer has dimension 6, which corresponds to the number of the QRE parameters to extract. To constrain the output belonging to the bounding box (7.4), an absolute *sine wave* activation layer after the output layer has been introduced. Specifically, it takes the prediction vector from the output layer, $\widehat{\boldsymbol{x}}_2$, and the constraint box $\boldsymbol{X}_2$ as input, and returns the predicted values $\boldsymbol{x}_2 \in \boldsymbol{X}_2$:

$$\boldsymbol{x}_2 = \text{SineWaveFunc}\,(\widehat{\boldsymbol{x}}_2, \boldsymbol{X}_2)$$

This is obtained by first normalizing $\widehat{\boldsymbol{x}}_2$ to the interval of each quadrupolar parameter by dividing the difference between itself and the lower boundary by the range of the boundaries:

$$\widehat{\boldsymbol{x}}_2 \leftarrow \frac{(\widehat{\boldsymbol{x}}_2 - \boldsymbol{L})}{(\boldsymbol{U} - \boldsymbol{L})} \tag{7.12}$$

where $\boldsymbol{L}$, and $\boldsymbol{U}$ represent the lower and the upper boundary vectors respectively. Then, the absolute value of the sine function is applied to the normalization result:

$$\widehat{\boldsymbol{x}}_2 = |\sin(\widehat{\boldsymbol{x}}_2)|$$

Consequently, the vector is re-scaled in the corresponding expected intervals by the inverse operation of (7.12):

$$\boldsymbol{x}_2 \leftarrow (\widehat{\boldsymbol{x}}_2 \odot (\boldsymbol{U} - \boldsymbol{L})) + \boldsymbol{U}$$

The last step of the function is applied to the two final elements of the vector $\boldsymbol{x}_2$, which are the angular frequency position of the peaks on the NMRD profiles, i.e., ω_-, and ω_+. In particular, based on prior physical considerations, we want to ensure that the following condition is always satisfied

$$\omega_- < \omega_+$$

Therefore we apply a mask to maintain this condition and substitute the corrected values in the vector $\boldsymbol{x}_2$.

After the training, the neural network is *plugged* into the proposed iterative algorithm. The algorithm is referred to as *Plug and Play - Gauss-Seidel* (*PnP-GS*).

In the algorithm, at each iteration, the objective function is minimized with respect to each of the block coordinate vectors $\boldsymbol{x}_i$ over the subsets $\boldsymbol{X}_i$, $i = 1, 2$, as summarized in Algorithm 11.

The stopping condition is the relative distance between two successive values of the objective function:

$$g(\boldsymbol{x}_1, \boldsymbol{x}_2) = \|(\boldsymbol{K}_e \boldsymbol{x}_1 - (\boldsymbol{y} - \mathcal{F}_2(\boldsymbol{x}_2))\|_2^2 + \lambda \|\mathcal{D}_{n,0}\boldsymbol{x}_1\|_1 + \eta \|\mathcal{D}_{n,0}\boldsymbol{x}_1\|_2^2$$

i.e.,

$$\|g(\boldsymbol{x}_1^{(k)}, \boldsymbol{x}_2^{(k)}) - g(\boldsymbol{x}_1^{(k-1)}, \boldsymbol{x}_2^{(k-1)})\|_2 \leq \text{Tol}\|g(\boldsymbol{x}_1^{(k)}, \boldsymbol{x}_2^{(k)})\|_2 \quad , \qquad \text{Tol} > 0 \tag{7.13}$$

Algorithm 11 PnP-GS

1: Set $k = 0$, and choose a starting guess $(\boldsymbol{x}_1^{(0)}, \boldsymbol{x}_2^{(0)})$.
2: **repeat**
3: $k = k + 1$
4: **Linear parameters update.** By using the truncated Newton interior-point method, compute

$$\boldsymbol{x}_1^{(k)} \in \arg\min_{\boldsymbol{z} \in \boldsymbol{X}_1} g(\boldsymbol{z}, \boldsymbol{x}_2^{(k)})$$

5: **Quadrupolar parameters update.** By using the trained neural network ν_Θ, compute the prediction

$$\boldsymbol{x}_2^{(k)} = \underset{\boldsymbol{z} \in \boldsymbol{X}_2}{\nu_\Theta} \left(g(\boldsymbol{x}_1^{(k)}, \boldsymbol{z}) \right)$$

6: **until** `the stopping condition is satisfied` (7.13)
7: **return** $(\boldsymbol{f}, R_0) = \boldsymbol{x}_1^{(k)}$ and $\boldsymbol{\psi} = \boldsymbol{x}_2^{(k)}$ ▷ Result $(\boldsymbol{f}, R_0, \boldsymbol{\psi})$

The PnP-GS algorithm, as described before, is a variant of the two-block nonlinear Gauss-Seidel method used in the previous Chapter 6 [39, 155]. A critical property of the traditional Gauss-Seidel method is the monotonic decrease of the objective function g at each iteration, i.e.,

$$g(\boldsymbol{x}_1^{(k)}, \boldsymbol{x}_2^{(k)}) \leq g(\boldsymbol{x}_1^{(k)}, \boldsymbol{x}_2^{(k-1)}) \leq g(\boldsymbol{x}_1^{(k-1)}, \boldsymbol{x}_2^{(k-1)})$$

In the proposed PnP-GS algorithm, we replace the update of the second block $\boldsymbol{x}_2$ with a prediction from a trained neural network, ν_Θ.

Since ν_Θ is a learned predictor, it does not inherently enforce the descent condition:

$$g(\boldsymbol{x}_1^{(k)}, \boldsymbol{x}_2^{(k)}) < g(\boldsymbol{x}_1^{(k)}, \boldsymbol{x}_2^{(k-1)}) \tag{7.14}$$

In contrast to explicit optimization steps, the neural network's output cannot be assumed to guarantee a monotonic decrease in the objective function at every iteration.

The adopted approach in this proposed method is to verify the decreasing behavior of the algorithm through empirical observation. In practice, the evolution of the objective function g is monitored throughout the iterative process. Empirical evidence indicates that, in the vast majority of iterations, the condition (7.14) is satisfied. Hence, although the neural network update does not intrinsically guarantee a monotonic descent of the objective function, the overall iterative scheme mimics the convergence behavior analogous to that of the traditional Gauss-Seidel method. This empirical control provides evidence of the robustness and reliability of the proposed approach.

7.3 Numerical Results and Discussion

In this section, the results produced by the proposed framework when applied to food samples of two distinct types of cheese, effectively illustrating typical test scenarios are presented and discussed. Specifically, the following samples have been considered (taken from [26, 16]):

- [PC]: A sample of Pecorino cheese produced from livestock raised in a region characterized by brown soil.

- [PR]: A sample of 24-month aged Parmigiano - Reggiano cheese.

Details on the generation of the training set, the training procedure, and the error metrics used for evaluation are provided in sections 7.3.1 and 7.3.2. The experiments involving analysis of the loss function in equation (7.10) are discussed in 7.3.3. Finally, in 7.3.4, a comprehensive report and discussion of the algorithm's performance are presented, comparing the results obtained with those produced by the Model-FreeFFC Matlab software [168] based on the *AURORA* algorithm described in the previous Chapter (6).

The numerical computations were performed using Matlab R2023b on a laptop with a 2 GHz Quad-Core Intel Core i5 processor and 16 GB of 3733 MHz RAM. The neural network was implemented using the Deep Learning Toolbox. Also in this result section, the frequencies f[1] are considered instead of angular frequencies ω, using the relationship $\mathrm{f} \equiv \omega/(2\pi)$.

7.3.1 Training Procedure

The neural network ν_Θ, which takes into input the quadrupolar part of the NMRD signal R^{HN}, has been trained using a training set composed of synthetic signals. Specifically, signals were obtained by randomly sampling sets of the quadrupolar parameters from uniform and continuous distributions in the $\mathcal{B}_{\boldsymbol{\psi}}$ range and using the model (7.2).

The values $\bar{C}$, and $\bar{\tau}$ in the constraints set $\mathcal{B}_{\boldsymbol{\psi}}$ (7.4) are set equal rounding up to the nearest integer of the maximum value between the correspondent values for both samples. The interval $[\omega_\ell, \omega_u]$ in (7.4), which represents the region where R_1 interrupts its decaying due to the quadrupolar relaxation effect, is defined by inspection of the NMRD profile and rounding up to the nearest integer of the minimum and maximum value of both samples. The values for the angles Θ, and Φ, which represent the orientation of the molecular system, were constrained within the interval $[0, 1]$. This range was selected to cover the full possible span of angular values, providing flexibility for the network to adapt to the varied quadrupolar interactions present in the cheese samples without bias towards a specific orientation. In Table 7.1 the range of $\mathcal{B}_{\boldsymbol{\psi}}$ is collected.

Table 7.1: Range of the parameters, $\mathcal{B}_{\boldsymbol{\psi}}$, selected to generate the training set.

Variable	Interval
C^{HN} $[\mu s/s^2]$	$[0, 8]$
Θ [rad]	$[0, 1]$
Φ [rad]	$[0, 1]$
τ_Q $[\mu s]$	$[0, 3]$
$[\omega_\ell, \omega_u]$ $[\mathrm{rad}/\mu s]$	$[10, 20]$

Figure 7.2 presents a portion of the training set, where several signals are plotted. It is worth noting that the peaks are shifted in position and exhibit varying amplitudes, depending on the randomly sampled values used to generate each corresponding signal, capturing the diversity within the synthetic dataset.

In Table 7.2, the training characteristics are specified. Moreover, ADAM optimizer [105] has been used to minimize the loss function. The network training has been performed several times, varying the value of α in the quadrupolar loss function (7.10). Starting from a very small value, $1 \cdot 10^{-3}$, which corresponds to an almost null influence of the profile fitting in the minimization of the loss function, to 1, which corresponds to not taking into account the prediction error by the network. The following α values have been used: $\alpha = \{1 \cdot 10^{-3}, 1 \cdot 10^{-1}, 3 \cdot 10^{-1}, 5 \cdot 10^{-1}, 7 \cdot 10^{-1}, 9 \cdot 10^{-1}, 9.99 \cdot 10^{-1}, 1\}$.

[1] The notation f is used to refer to frequencies instead of the usual ν to avoid confusion with the neural network, ν_θ. Moreover, it is worth pointing out that the notation f differs from f. The latter refers to the correlation time distributions and it is always presented as a function of the correlation times, i.e., $f(\tau_c)$.

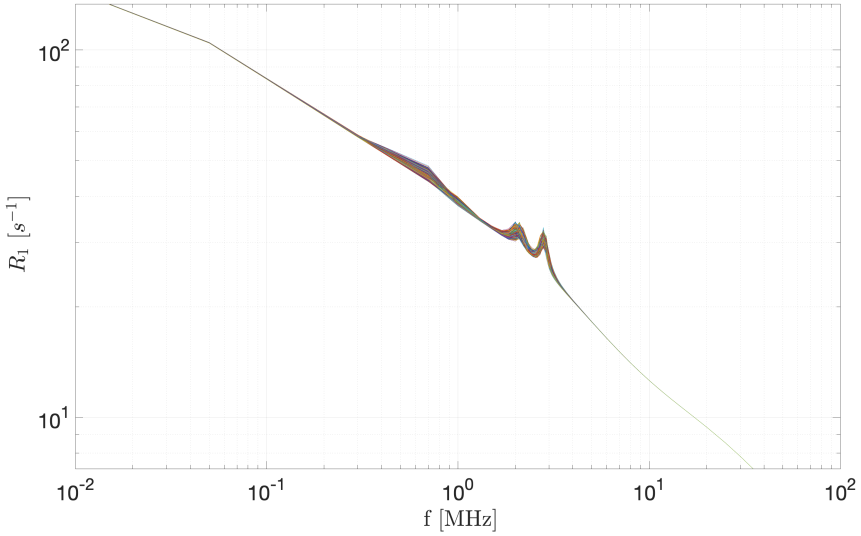


Figure 7.2: A subset of synthetic signals in the training set, generated randomly sampling $\boldsymbol{\psi}_m \in \mathcal{B}_{\boldsymbol{\psi}}$, $m = 1, \ldots, 10^4$ from a uniform and continuous distribution.

Table 7.2: Training parameters selected to train ν_Θ.

Dimension of the training set	10^4
Computational time for a single training $[s]$	10^3
Number of epochs	10^3
Learning rate	10^{-4}

7.3.2 Testing Procedure

Next, the method used to test the neural network is detailed. This includes the initial parameter guesses, stopping criteria, and the computational setup for the numerical tests.

As for the previous algorithm (*AURORA* 10), the starting guess for the parameter $\psi_1^{(0)} \equiv C^{HN}$ is obtained from the literature [4]:

$$C^{HN} = \frac{2}{3}\left(\frac{\mu_0}{4\pi}\frac{\gamma_H\gamma_N\hbar}{r_{HN}^3}\right)^2 \approx 0.18\left[\frac{\mu s}{s^2}\right] \tag{7.15}$$

with the values of physical constants reported in Table 6.1 of the previous Chapter 6.

Considering the other quadrupolar parameters, the starting values are reported in Table 7.3.

The value of the tolerance parameters used in the stopping criteria of the algorithms is $\text{Tol} = 10^{-2}$, while the maximum number of 10^3 iterations k has been set, but never reached.

The computational cost of the algorithm is evaluated in terms of execution time, which encompasses the duration required for processing the testing dataset. The number of iterations needed to reach the stop condition has been also considered in terms of efficiency. By analyzing both execution time and iteration count, insights into the efficiency and scalability of the neural network can be gained, informing its practical applicability in real-world scenarios.

Table 7.3: Starting guess selected for the quadrupolar parameters.

Starting guess	Parameter correspondent	Value
$\psi_1^{(0)}$	C^{HN} $[\mu s/s^2]$	0.18
$\psi_2^{(0)}$	$\sin^2(\Theta)$ [rad]	1/2
$\psi_3^{(0)}$	$\sin^2(\Phi)$ [rad]	1/2
$\psi_4^{(0)}$	τ_Q $[\mu s]$	1
$\psi_5^{(0)}$	ω_ℓ [rad/μs]	12
$\psi_6^{(0)}$	ω_u [rad/μs]	18

Error Metrics

To evaluate the performance of the network during training and testing, specific error metrics have been used. The fitted NMRD profiles, computed by the Algorithm 11, and *AURORA* (10), are compared to the experimental acquired R_1 employing the χ^2 value defined as follows:

$$\chi^2 = \sum_{i=1}^{m} \frac{(R_{1_i}^{\text{est}} - R_{1_i})^2}{(m-1)} \tag{7.16}$$

where $\boldsymbol{R}_1^{\text{est}}$ is the estimated data value, i.e.,

$$\boldsymbol{R}_1^{\text{est}} = \boldsymbol{K}\widetilde{\boldsymbol{f}} + \widetilde{R}_0 + \mathcal{F}_2(\widetilde{\boldsymbol{\psi}})$$

where $(\widetilde{\boldsymbol{f}}, \widetilde{R}_0, \widetilde{\boldsymbol{\psi}})$ are the computed parameters.

To obtain a global parameter that accounts for both fit and parameter error predictions to select the best value for α in the training step, the following L_2 squared norm-based metric computed as follows has been proposed:

$$\mathcal{E}_\alpha = \frac{\|\boldsymbol{R}_1 - \boldsymbol{R}_1^{\text{est}}\|_2^2}{\|\boldsymbol{R}_1\|_2^2} + \frac{\|\boldsymbol{f} - \boldsymbol{f}^{\text{est}}\|_2^2}{\|\boldsymbol{f}\|_2^2} + \frac{\|R_0 - R_0^{\text{est}}\|_2^2}{\|R_0\|_2^2} + \frac{\|\boldsymbol{\psi} - \boldsymbol{\psi}^{\text{est}}\|_2^2}{\|\boldsymbol{\psi}\|_2^2} \tag{7.17}$$

The proposed metric comprehensively accounts not only for the relative error in the network's predictions but also incorporates the fitting error and the error associated with the linear components. This holistic approach ensures a more robust and accurate evaluation of the model's performance.

Finally, as in the work presented in Chapter 5, the computed correlation time distribution $\boldsymbol{f}$ extracted by the two algorithms have been quantitatively compared determining the peak values and the area below $\boldsymbol{f}$ in the neighbourhood of such peaks, defining such a value as SpecificWeight (5.19).

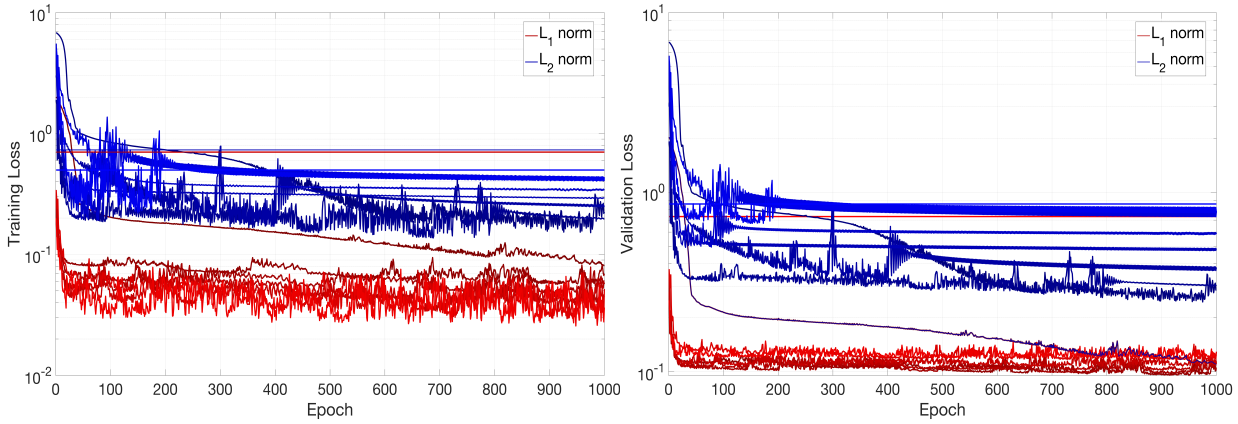
Let us briefly recall the formula for the SpecificWeight:

$$\text{SpecificWeight}_\ell = \sum_{j=1}^{n_\ell} \tau_{c_j} f(\tau_{c_j}) \quad , \qquad \tau_{c_j} \in \mathcal{I}_\ell$$

where n_ℓ is the number of correlation times inside the neighborhood of interest $\mathcal{I}_\ell$ (using the Full Width at Half Maximum parameter), and with $\ell = 1, \dots, n_p$, number of local maxima in $\boldsymbol{f}$.

7.3.3 Model Loss and Trained Network Selection

The choice of the model loss has been validated by training experiments. Specifically, the network performance has been analyzed by varying both the considered norm ($p = 1$ for the L_1-norm and $p = 2$ for the L_2-norm in (7.10)) and the parameter $\alpha \in [0, 1]$ in (7.11). In this paragraph, the results related to this analysis are shown.

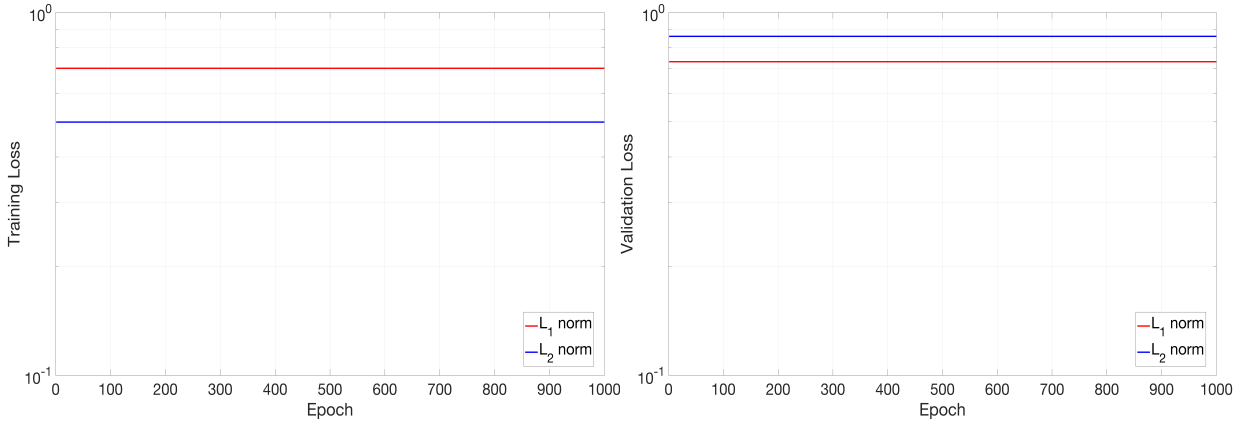


(a) Training loss. (b) Validation loss.

Figure 7.3: In red the expression for the model loss with $p = 1$, while in blue the expression with $p = 2$. The different curves refer to different values of α.

In Figure 7.3, the training and validation loss evolution are respectively shown. As evident from the figure, the network trained with the L_1 norm exhibits a steeper decline in both training and validation losses compared to the L_2 norm case. In both cases, the loss functions rapidly decrease in the initial epochs and continue to decrease consistently throughout the training period for the L_1 norm case. From the training loss evolution, this suggests that this norm facilitates faster convergence during training compared to L_2; while from the validation loss evolution, this suggests that the L_1 norm enhances the generalization capability of the model.

The only exception is represented for the case of $\alpha = 1$, where the loss function evolutions are constant, meaning that the network is not learning. These cases represent the cases where the network tries to learn only from the NMRD fitting and it doesn't take into account the error on predictions (see (7.11) with $\alpha = 1$), and it is an expected result. For clarity, in Figure 7.4 only this case is shown.



(a) Training loss. (b) Validation loss.

Figure 7.4: Case model losses with $\alpha = 1$, i.e., the network during the training only takes into account the error on NMRD profile fitting, while it does not consider the error on the quadrupolar predictions. In red the expression for the model loss with $p = 1$, while in blue the expression with $p = 2$.

7.3.4 Results for Cheese Samples

In this part, the results obtained by applying the two algorithms (PnP-GS, and AURORA) to NMRD real profiles obtained by two experimental samples, already discussed at the beginning of this section are presented. The Pecorino cheese sample is referred to as *PC*, while the Parmigiano cheese sample to as *PR*. Regarding the samples, the PC sample has been produced from livestock raised in a region characterized by brown soil; while the PR sample is a 24-month-aged Parmigiano-Reggiano cheese. These samples were selected to illustrate the impact of varying compositional and environmental factors relevant to the food industry in the experimental analysis.

The R_1 data for both samples are measured at 45 frequency values f, ranging from 10^{-2} to 10^{1} MHz. The error intervals for the PR case vary from ± 0.1 to ± 4.4, while for the PC case from ± 0.4 to ± 4.1. These experimental signals are illustrated in Figure 7.5, where in blue and black are plotted the PR and the PC samples respectively.

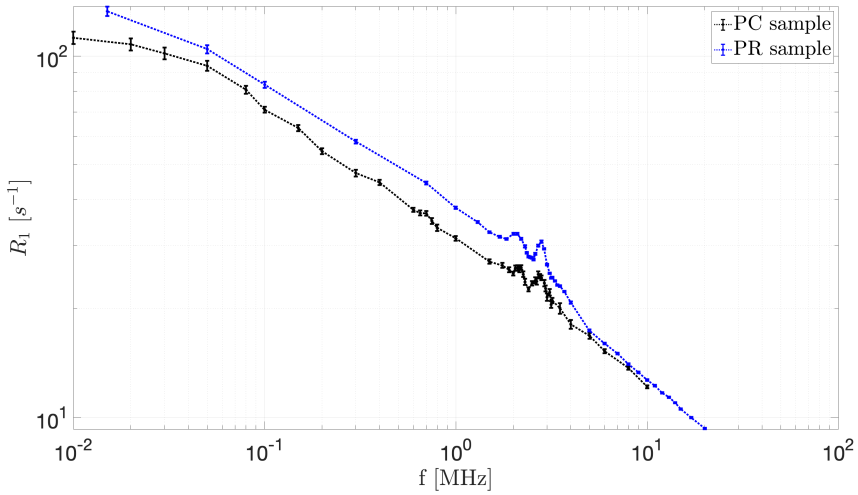


Figure 7.5: Test set: NMRD real profiles with error intervals from cheese samples. Specifically, in black is the PC sample, while in blue is the PR sample.

In the case of the proposed integrated method, the numerical tests were performed by using the best result from the model loss selection training procedure shown in Figure 7.3, i.e., $p = 1$, and by varying the hyper-parameter α. The global quality of the method is evaluated using the metrics described in the paragraph 7.3.2, showing the $\mathcal{E}_\alpha$, and the χ^2 evolution varying α for both samples. Finally, the NMRD profile fit and the correlation time distributions for the best α found are presented.

It is important to emphasize that the neural network was never exposed to the experimental signals from the test set during training.

Pecorino Cheese (PC) Sample

The following results refer to the first analyzed sample, i.e., the Pecorino cheese (PC) sample. In Figure 7.6 the proposed method performances are shown in terms of $\mathcal{E}_\alpha$ (7.17) varying α in the model loss. The best result obtained is at $\alpha = 0.7$, showing the importance of taking into account both the error on the predictions, and the error on NMRD fitting during the network training. In Table 7.4, the values of $\mathcal{E}_\alpha$ and the last value of χ^2 computed by the algorithm are indicated for each value of α. Figure 7.7 shows the evolution of the χ^2 values in the outer iteration loop (regularization parameter update) for each value of α. It is shown that

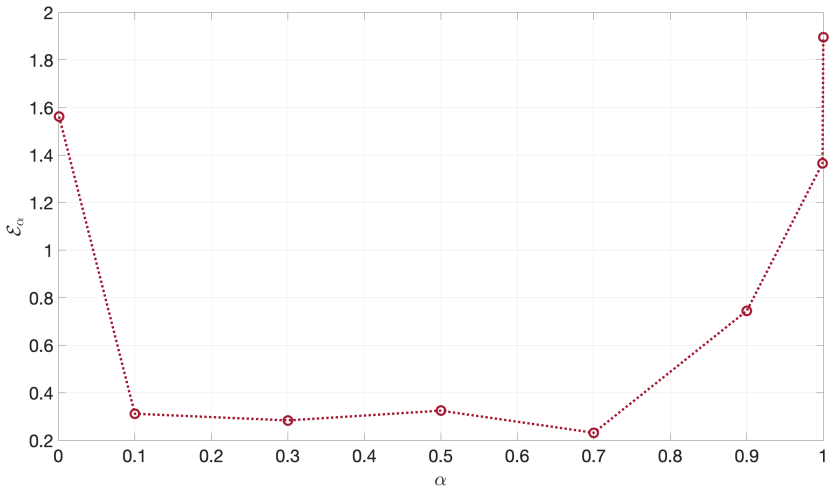


Figure 7.6: PC sample analysis. $\mathcal{E}_\alpha$ evolution varying α in the model loss function during the network training.

Table 7.4: PC sample analysis. $\mathcal{E}_\alpha$ and last value of χ^2 varying α in the model loss function during the network training.

	$\alpha = 0.001$	$\alpha = 0.1$	$\alpha = 0.3$	$\alpha = 0.5$	$\alpha = 0.7$	$\alpha = 0.9$	$\alpha = 0.999$	$\alpha = 1$
$\mathcal{E}_\alpha$	1.562	0.312	0.284	0.325	0.232	0.745	1.365	1.896
χ^2 (last)	0.630	0.518	0.497	0.554	0.479	0.575	0.591	0.806

the minimum value for χ^2 at the final iteration is achieved for $\alpha = 0.7$, confirming the observation made during the training phase.

Additionally, it is important to note the worst results occur for $\alpha = 1$. In this case, the χ^2 values are higher and remain constant compared to other cases. This outcome aligns with the network training results for $\alpha = 1$ (see Figure 7.4), where the training loss remains constant, indicating no learning. Incorporating this poorly trained network into the proposed algorithm results in suboptimal predictions, affecting both the linear part solved by numerical methods and the non-linear part addressed by the neural network. The fitted profile presented in Figure 7.8 (a) shows in black line the data and error bars, while the fitted curves are represented in red line for the case of the proposed PnP-GS algorithm and in green line for the case of *AURORA*. The zoom in the frequencies of the QRE interval is shown in Figure 7.8 (b). Qualitatively, it is evident the excellent agreement between the two methods, showing that the proposed algorithm can perfectly fit the experimental data. The predictions obtained by the two algorithms and the error metrics are presented in Table 7.5, showing a perfect agreement between the two methods. Moreover, in terms of χ^2, the PnP-GS algorithm performs better, with a lower value compared to *AURORA*. An interesting result is the efficiency of the algorithm that uses the neural network to extract the quadrupolar parameters compared to the *AURORA* algorithm. Hence, measuring the efficiency in terms of the number of iterations and computational time, it is shown that the PnP-GS algorithm requires an order of magnitude fewer iterations and significantly less computational time than the *AURORA* algorithm (after training).

Concerning the extracted correlation time distributions by the two methods, in Table 7.6, a perfect agreement among the two algorithms in locating the peak at shortest correlation time, $\tau_c = 1.87 \cdot 10^{-2}$ μs is observed. The algorithms present a quite good agreement at the other two correlation times, especially at the longest one. This result is qualitatively presented in Figure 7.9, showing that all two methods agree

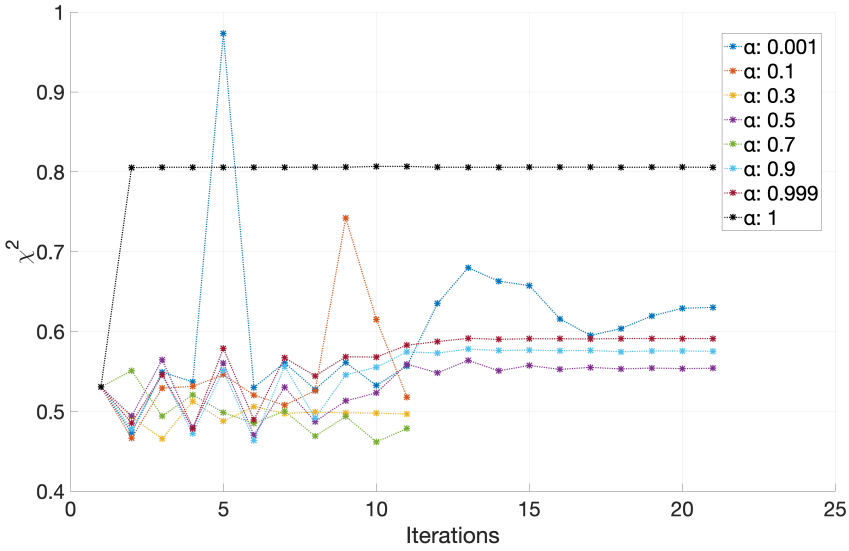
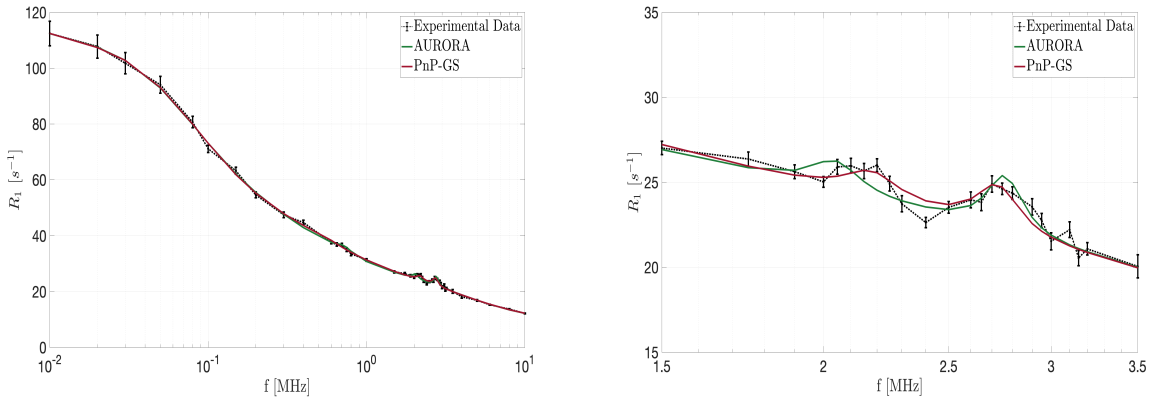


Figure 7.7: PC sample analysis. χ^2 evolution varying α in the model loss function during the network training.



(a) Full profile.

(b) Zoom of the profile in QRE interval.

Figure 7.8: PC sample analysis. The NMRD data and error bars are in black, while the fitted curves by AURORA, and by PnP-GS are in green and red respectively.

concerning the main features of the PC sample's distribution.

Parmigiano-Reggiano Cheese (PR) Sample

The analysis of the Parmigiano-Reggiano cheese sample follows. The evolution of the error metrics is provided and the obtained results are compared with those obtained using *AURORA*.

The method performance in terms of $\mathcal{E}_\alpha$ is presented in Figure 7.10, showing the same result obtained for the PC sample, i.e., the minimum value is obtained at $\alpha = 0.7$. The numerical values of $\mathcal{E}_\alpha$ and the last value of χ^2 varying α are specified in Table 7.7. The χ^2 evolution is shown in Figure 7.11, confirming that the best algorithm result is obtained using the trained network with $\alpha = 0.7$ in the model loss function. The NMRD profile fitting in the case for the best value obtained ($\alpha = 0.7$) is shown in Figure 7.12 (the full profile

Table 7.5: PC sample analysis. Predictions obtained by PnP-GS, and by AURORA.

	PnP-GS	**AURORA**
R_0	8.968	8.930
C^{NH}	2.753	2.014
θ	1.049	1.061
ϕ	0.9735	1.085
τ_Q	1.391	1.266
ν_-	2.043	2.178
ν_+	2.762	2.727
χ^2	0.497	0.755
Number of iterations	20	122
Computational time [s]	10	180

Table 7.6: PC sample analysis. Position (τ_c) and amplitude $f(\tau_c)$ of the distribution peaks sorted by $f(\tau_c)$.

Algorithm	τ_c [μs]	$f(\tau_c)$	Half-width	SpecificWeight
AURORA	0.9770	55.2006	0.2397	108.7355
	0.1353	36.5104	0.0158	36.5484
	0.0187	24.8273	0.0023	26.6269
PnP-GS	1.0975	99.8717	0.1307	108.7280
	0.1707	24.6778	0.0349	36.5321
	0.0187	14.7087	0.0045	26.6457

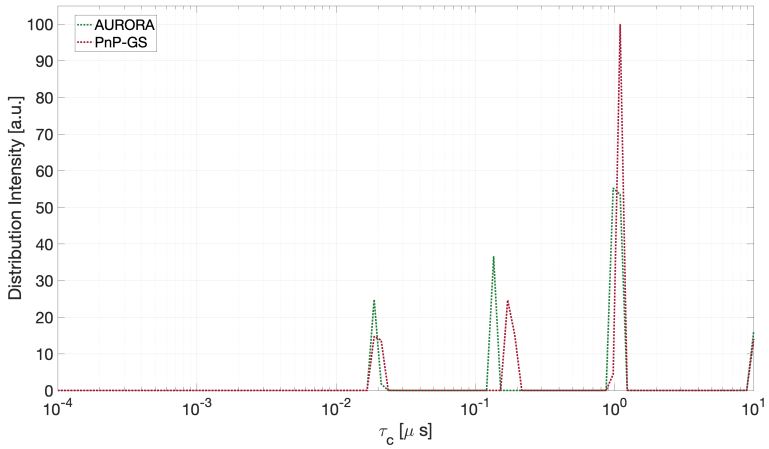


Figure 7.9: PC sample analysis. Correlation time distribution $f(\tau_c)$ extracted by AURORA (green lines), and by PnP-GS (red lines).

(a), while the zoom in QRE interval (b)). This result confirms the excellent agreement between the two methods and the experimental data. Also, in this case, a perfect agreement between the predictions of the two methods is obtained (Table 7.8). PnP-GS is faster and more efficient, as is shown from the χ^2 values of the two methods, as well as from the computational time and the number of iterations. Finally, considering

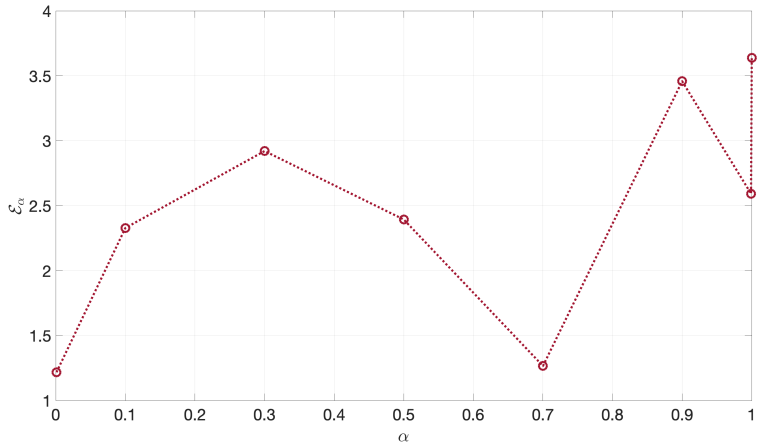


Figure 7.10: PR sample analysis. $\mathcal{E}_\alpha$ evolution varying α in the model loss function during the network training.

Table 7.7: PR sample analysis. $\mathcal{E}_\alpha$ and last value of χ^2 varying α in the model loss function during the network training.

	$\alpha = 0.001$	$\alpha = 0.1$	$\alpha = 0.3$	$\alpha = 0.5$	$\alpha = 0.7$	$\alpha = 0.9$	$\alpha = 0.999$	$\alpha = 1$
$\mathcal{E}_\alpha$	1.217	2.687	2.314	2.389	1.236	2.974	3.335	3.578
χ^2 (last)	0.191	0.388	0.228	0.233	0.120	0.446	0.776	1.328

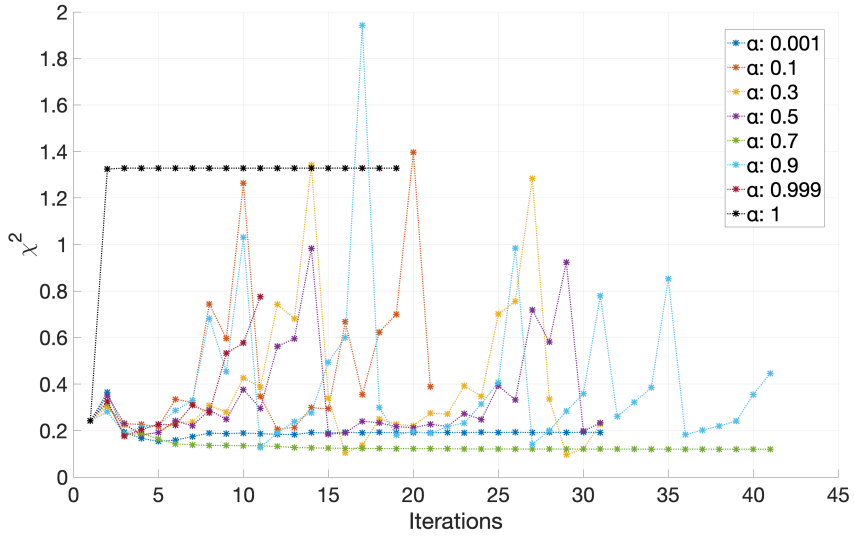
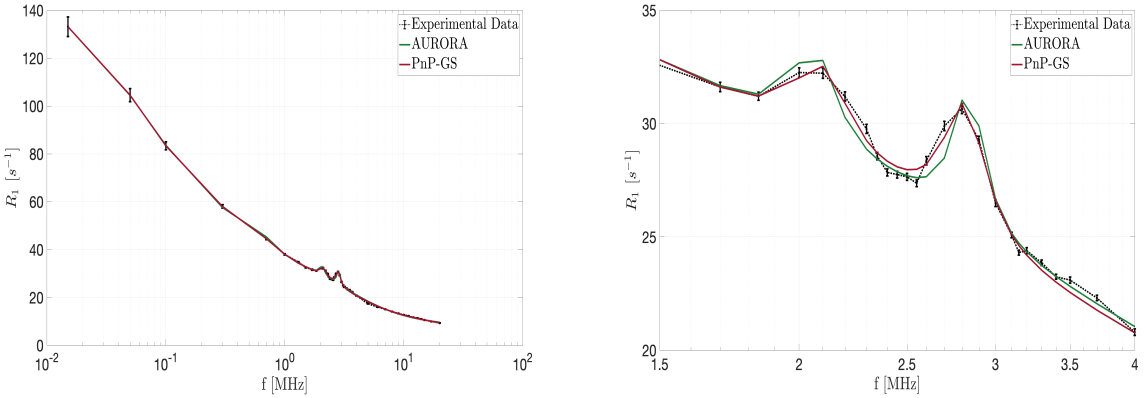


Figure 7.11: PR sample analysis. χ^2 evolution varying α in the model loss function during the network training.

the correlation time distribution analysis in the PR case, *AURORA* identifies more peaks compared to the PnP-GS algorithm. Qualitatively, from Figure 7.13, the two peaks at lower correlation times and the one at the longest correlation time show a good agreement. Concerning the second longer, PnP-GS finds a peak at $\tau_c = 0.2719$ μs, while *AURORA* splits that peak into two at $\tau_c = 0.1717$ μs, and $\tau_c = 0.6893$ μs. The mismatch is related to the ill-posedness of the mathematical problem.



(a) Full profile.

(b) Zoom of the profile in QRE interval.

Figure 7.12: PR sample analysis. The NMRD data and error bars are in black, while the fitted curves by AURORA, and by PnP-GS are in green and red respectively.

Table 7.8: PR sample analysis. Predictions obtained by PnP-GS, and by AURORA.

	PnP-GS	**AURORA**
R_0	3.832	3.290
C^{NH}	5.784	5.664
θ	0.989	1
ϕ	1.046	0.859
τ_Q	1.386	1.022
ν_-	2.065	2.096
ν_+	2.839	2.815
χ^2	0.120	1.189
Number of iterations	40	189
Computational time [s]	9	45

Table 7.9: PR sample analysis. Position (τ_c) and amplitude $f(\tau_c)$ of the distribution peaks sorted by $f(\tau_c)$.

Algorithm	τ_c [μs]	$f(\tau_c)$	Half-width	SpecificWeight
AURORA	0.1707	43.0202	0.0202	44.3755
	1.7475	22.1196	0.7016	85.2641
	0.0266	19.4388	0.0058	38.1599
	0.6893	9.5693	0.2278	34.3516
	0.0023	8.6636	$3.4447{\cdot}10^{-4}$	11.6631
	0.0010	$6.0052{\cdot}10^{-4}$	$7.3211{\cdot}10^{-5}$	0.0017
PnP-GS	1.9630	71.5316	0.3623	67.1817
	0.2719	55.7332	0.0363	0.4441
	0.0236	24.1669	0.0053	38.1577
	0.0026	3.0452	$5.3259{\cdot}10^{-4}$	11.6695

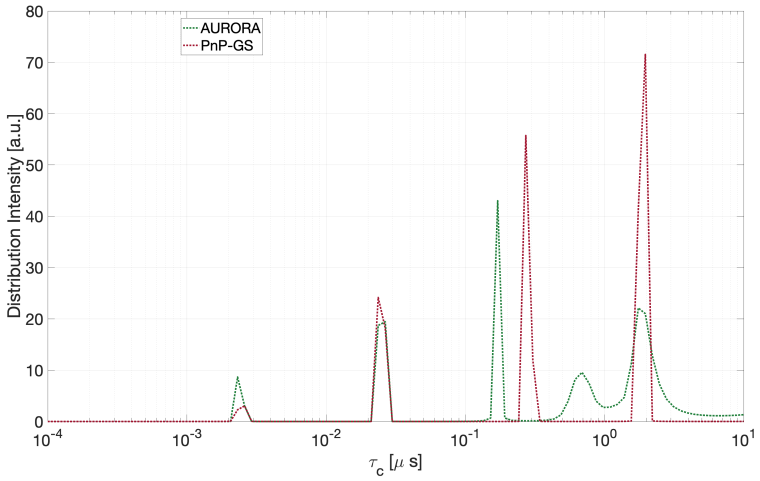


Figure 7.13: PR sample analysis. Correlation time distribution $f(\tau_c)$ extracted by *AURORA* (green lines), and by PnP-GS (red lines).

The observed results demonstrate the construction of a PnP neural network capable of returning the quadrupolar QRE parameters for two different types of cheese. Embedding this network in software to fit the NMRD profiles makes its use more robust and effective. These preliminary results form the basis for large-scale application. By enriching the training set with appropriately measured samples, it will be possible to proceed with the automatic identification of the characteristics of the examined samples.

Conclusions

This thesis has provided a comprehensive investigation into advanced methodologies for analyzing Nuclear Magnetic Relaxation Dispersion (NMRD) profiles. The work focused on designing, developing, and validating innovative algorithms tailored to address key challenges in this domain. The presented approaches to this application encompass traditional numerical techniques, regularization-based optimization, and neural network-based methods. These contributions have demonstrated significant potential to enhance the toolkit available to researchers for characterizing molecular dynamics in a range of materials and application scenarios. Below, the key contributions of each chapter of the second part of this thesis are summarized in detail.

In Chapter 5, a comparative study was conducted to evaluate three algorithms based on different regularization strategies, i.e., *MF-MUPen*, based on locally adapted L_2 regularization, *MF-L1*, based on L_1 regularization, and *MF-MUPen*, based on multi-penalty regularization, consisting of local-L_2, and L_1 penalties. These three methods were proposed to analyze NMRD profiles derived from experimental data.
The study was performed using two datasets representative of different applicative scenarios. All three algorithms demonstrated consistency in identifying primary peak positions, indicating fundamental robustness in capturing the core characteristics of the samples. Among these, the *MF-MUPen* algorithm stood out for its robustness in the presence of noisy data. However, differences in the number of peaks detected and their respective amplitudes highlighted variations in the sensitivity and adaptability of each algorithm. This analysis underscores that while all methods are effective in their core task, the choice of algorithm should be guided by the specific noise levels and sensitivity requirements of the intended application [154]. These findings would allow researchers to make more precise and context-specific decisions when designing and executing practical experimental setups to investigate the molecular dynamics of different samples.

In Chapter 6, a strategy to address a more complex challenge of analyzing NMRD profiles is presented. Specifically, the focus is on profiles influenced by the Quadrupolar Relaxation Enhancement. This introduced a nonlinear model, requiring the solution of a constrained nonlinear least-squares problem. To tackle this, a novel methodology was proposed, integrating a model-free approach with L_1 regularization. The optimization problem was effectively solved using a two-block nonlinear Gauss-Seidel method, with theoretical guarantees established for the existence of a solution and the convergence of the iterative process to critical points. Furthermore, following the Balancing Principle introduced by Ito, an automatic update rule for the regularization parameter is proposed, ensuring adaptability and robustness across diverse scenarios. The resulting algorithm, *AURORA*, has been validated using first a synthetic dataset, and then by applying it to real data. The results demonstrated its ability to accurately estimate correlation time distributions and model the quadrupolar function.
Moreover, the algorithm can be viewed as a reference framework to construct parameter estimation procedures when the model parameters can be split into independent blocks allowing the use of different computational approaches for each block [155].
The *AURORA* algorithm has been included in the Matlab software tool `ModelFreeFFC` Tool for the inversion of NMRD profiles with QRE (available at

`https://site.unibo.it/softwaredicam/en/software/modelfree`).

Chapter 7 introduced a novel approach by developing the first neural network-based framework for analyzing NMRD profiles in the presence of QRE. Unlike traditional inverse methods presented in the previous chapters, this approach treated the problem as a black-box optimization task, achieving results consistent with those of validated numerical algorithms.

The neural network model was validated on real datasets and demonstrated superior computational speed once trained, offering a significant advantage for high-throughput industrial applications. However, it is important to note that this comparison does not take into account the training time required for the network, which remains manageable at approximately 10^3 seconds, even with a training set consisting of 10^4 signals. Once the network is trained and integrated into the algorithm, no retraining is necessary, enabling rapid processing of large datasets from various samples and allowing for accurate characterization of material properties. Unlike *AURORA*, which requires careful initialization for each application, the proposed algorithm offers a more streamlined, automated solution for efficiently processing real signals.

It demonstrates high accuracy in estimating correlation time distributions and quadrupolar parameters, reinforcing its potential for use in fast-paced industrial environments [167]. For instance, it can be applied to monitor ageing and ripening processes in cheeses and other fermented foods, analyze moisture content in dairy products, meats, and processed foods, and potentially predict shelf life or detect early signs of deterioration in fresh and packaged foods.

Despite its promising capabilities, the current approach has some limitations. First, the need for specialized and costly FFC-NMR equipment, which is not commonly available in standard food testing laboratories, may restrict its widespread use. Additionally, the accuracy of the neural network heavily relies on the quality and quantity of the training data. While the method was validated on specific cheese types using synthetic signals, applying it to other food products would require generating suitable training data, limiting its immediate applicability to a broader range of food matrices. However, the highly automated nature of the proposed method offers a significant advantage, as it reduces the need for manual intervention and expert knowledge, making it easier to adapt and scale once the appropriate data is available, potentially overcoming these limitations in the long term.

In conclusion, this thesis represents an advancement in the field of NMRD profile analysis. The algorithms and methodologies developed herein address critical challenges in robustness, adaptability, and computational efficiency. By bridging the gap between traditional numerical methods and modern machine learning techniques, this work contributes to the growing demand for accurate, efficient, and automated tools for molecular dynamics studies. The outcomes of this research not only advance the state-of-the-art but also lay the groundwork for future innovations, ensuring broad applicability across scientific and industrial domains.

Appendix A

Dispersion Analysis Applied to NMRD Profiles

The dispersion analysis presented in sections 5.3.3, and 6.3.3 has been carried out to investigate the robustness of the proposed methods and to extract the confidence interval of the computed parameters.

In this work, the analysis consisted of applying the algorithm of interest to a set of s artificial profiles obtained by adding to the curve uniformly distributed noise within the experimental error intervals. Generally, all the tests were performed setting $s = 500$.

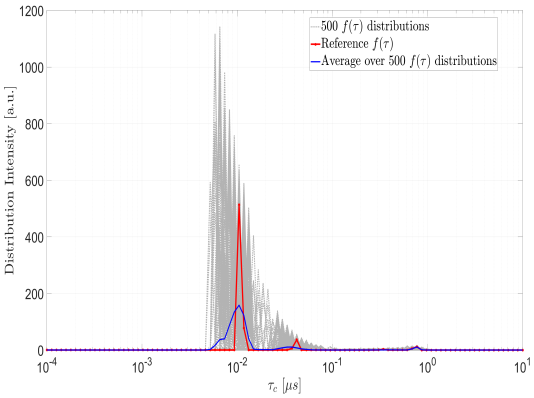
In this appendix, some results obtained by this analysis are shown both on samples presenting the QRE effect and not. The most important consequences required from this analysis were already discussed in the previous chapters, hence here the figures are shown for the sake of completeness.

Manganese Sample

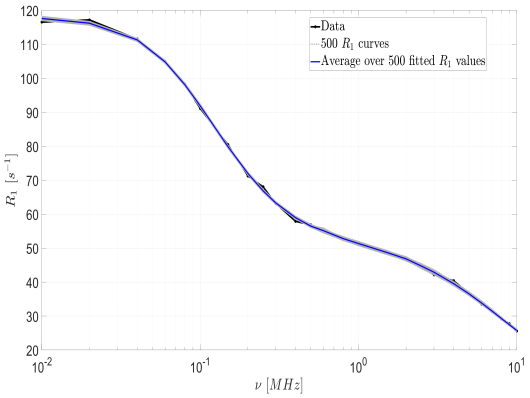
Referring to the manganese sample of Chapter 5, i.e., a case where the experimental profile does not present local maxima because the QRE effect does not occur, Figure A.1 shows the correlation time distributions (a) and the NMRD profiles (b) obtained by applying the MF-L1 Algorithm (7) of Chapter 5 to a set of 500 synthetic signals obtained starting from the real acquired one. Specifically, all the results for the 500 data are in light grey, the reference computed on the real data set is in red, and the mean values obtained averaging over the 500 set in blue. In Figure **??** the scatter plot representing the 500 computed R_0 parameters is shown, with the horizontal red line representing the mean value, i.e., $\bar{R}_0 = 1.19 \cdot 10^1 \ s^{-1}$ (reported in Table 5.1 of Chapter 5).

Parmigiano-Reggiano Sample

Here, the Parmigiano-Reggiano (PR) sample from Chapter 6, which presents the QRE effect, is analyzed. Figure A.3 displays the correlation time distributions (a) and NMRD profiles (b) obtained by applying AURORA to 500 synthetic signals derived from the real acquired data. The results for all 500 datasets are shown in light gray, with the reference computed on the real data in red and the mean values averaged over the set in blue. Figure A.4 presents the scatter plots for the R_0 (a), C^{HN} (b), and τ_Q (c) parameters, where the horizontal red line indicates the mean value. Similarly, Figures A.5 and A.6 show scatter plots for Θ, Φ, ν_-, and ν_+. For reference, the mean values and confidence intervals are summarized in Table A.1, recalling Table 6.8.



(a) 500 Computed correlation time distribution.



(b) 500 computed NMRD profiles.

Figure A.1: Manganese Sample. Computation of correlation time distributions and fitting profiles by MF-L1 to 500 data set. In (b) the real experimental data and the error bars are black.

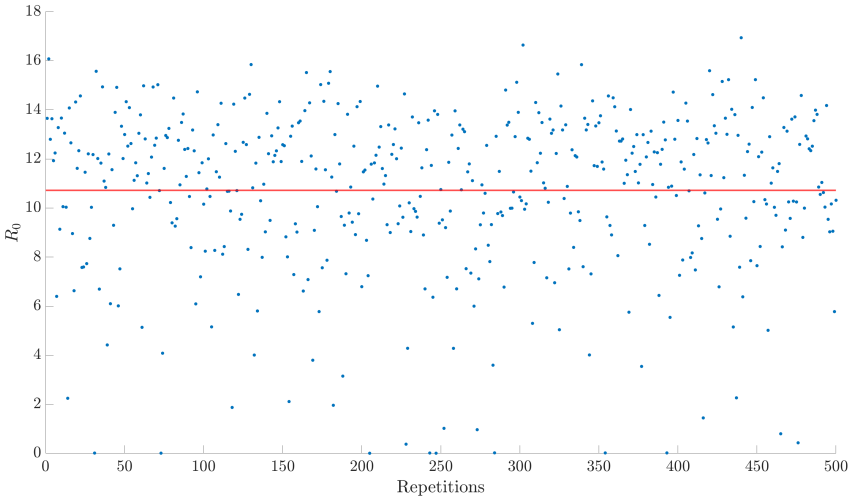
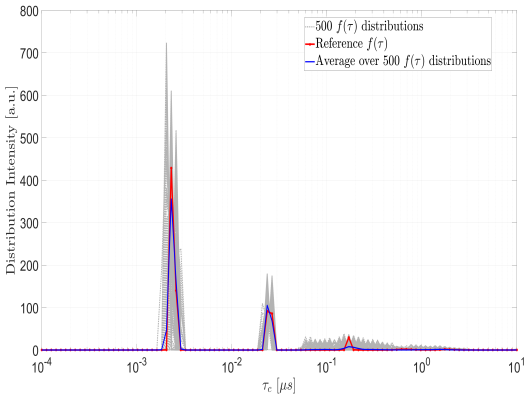


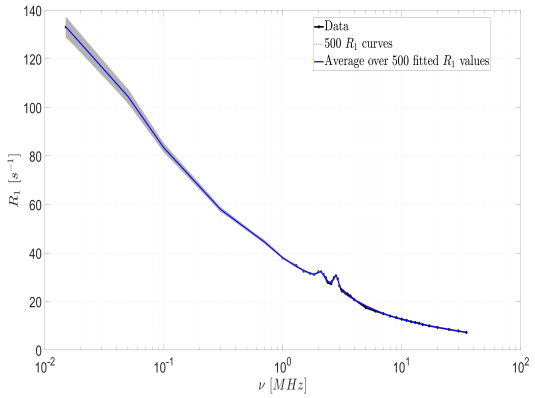
Figure A.2: Manganese Sample. Scatter plot reporting the 500 R_0 computed values by MF-L1.

Table A.1: Comparison of the Confidence Intervals, Mean, and Median for PR sample.

	Confidence Interval	Mean	Median
R_0 $[s^{-1}]$	[2.316, 4.013]	$3.308{\cdot}10^0$	$3.252{\cdot}10^0$
C^{HN} $[\mu s/s^2]$	[5.285, 6.351]	$5.736{\cdot}10^0$	$5.733{\cdot}10^0$
Θ[rad]	[1.058, 1.253]	$1.214{\cdot}10^0$	$1.253{\cdot}10^0$
Φ[rad]	[0.828, 0.891]	$0.860{\cdot}10^0$	$0.859{\cdot}10^0$
τ_Q $[\mu s]$	[0.937, 1.106]	$1.020{\cdot}10^0$	$1.020{\cdot}10^0$
ν_- [MHz]	[2.080, 2.111]	$2.096{\cdot}10^0$	$2.095{\cdot}10^0$
ν_+ [MHz]	[2.808, 2.821]	$2.815{\cdot}10^0$	$2.814{\cdot}10^0$

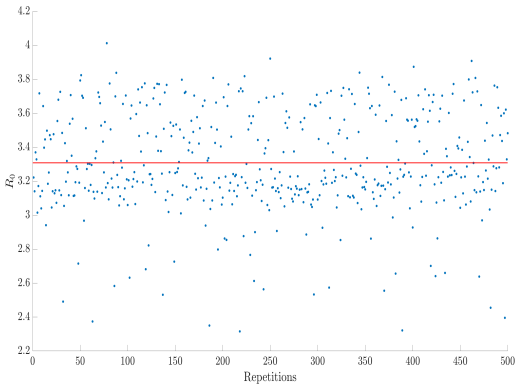


(a) 500 Computed correlation time distribution.

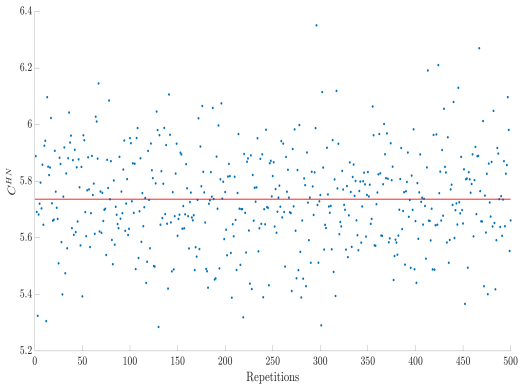


(b) 500 computed NMRD profiles.

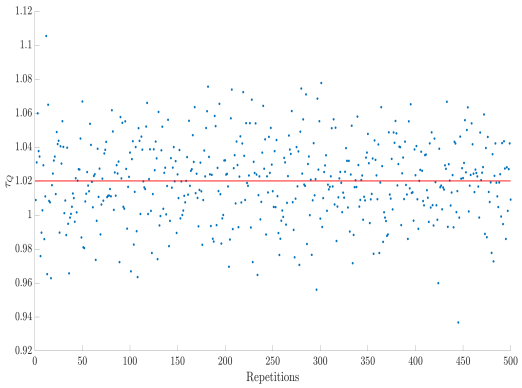
Figure A.3: PR sample. Computation of correlation time distributions and fitting profiles by AURORA to 500 data set. In (b) the real experimental data and the error bars are black.



(a) R_0.

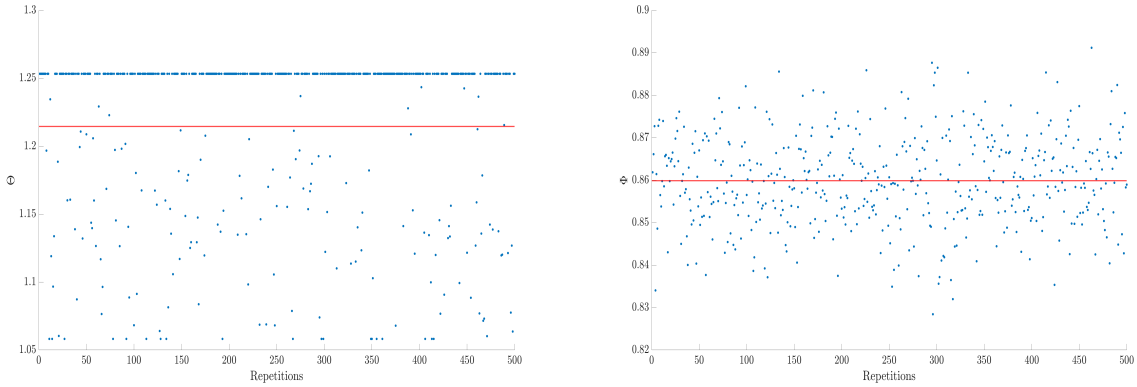


(b) C^{HN}.



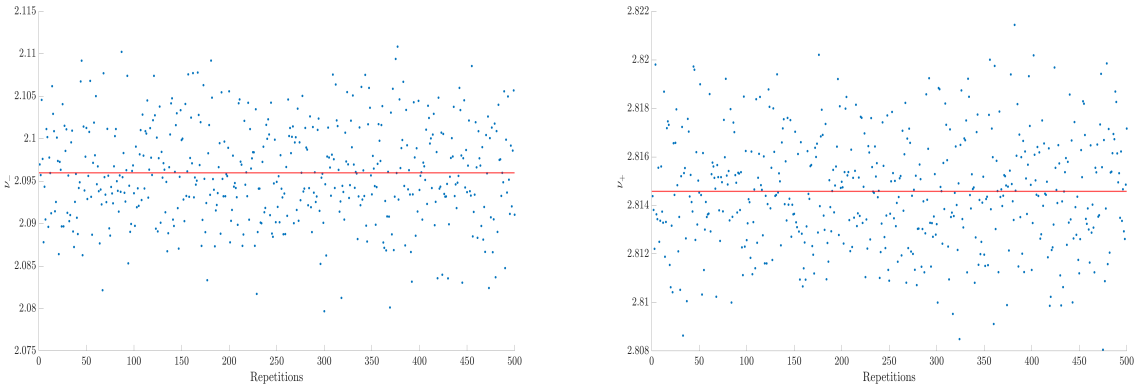
(c) τ_Q.

Figure A.4: PR sample. Scatter plot reporting the 500 computed values by AURORA.



(a) Θ. (b) Φ.

Figure A.5: PR sample. Scatter plot reporting the 500 computed values by AURORA.



(a) ν_-. (b) ν_+.

Figure A.6: PR sample. Scatter plot reporting the 500 computed values by AURORA.

Bibliography

[1] R. Kimmich and E. Anoardo. Field-cycling NMR relaxometry. *Progress in Nuclear Magnetic Resonance Spectroscopy*, 44(3):257–320, 2004.

[2] Pellegrino Conte. Applications of fast field cycling NMR relaxometry. In *Annual Reports on NMR Spectroscopy*, volume 104, pages 141–188. Elsevier, 2021.

[3] P. H. Fries and E. Belorizky. Simple expressions of the nuclear relaxation rate enhancement due to quadrupole nuclei in slowly tumbling molecules. *The Journal of Chemical Physics*, 143(4):044202, 2015.

[4] D. Kruk, E. Masiewicz, A. M. Borkowska, P. Rochowski, P. H. Fries, L. M. Broche, and D. J. Lurie. Dynamics of solid proteins by means of nuclear magnetic resonance relaxometry. *Biomolecules*, 9(11):652, 2019.

[5] T. Jeoh, N. Karuna, N. D. Weiss, and L. G. Thygesen. Two-dimensional 1H-nuclear magnetic resonance relaxometry for understanding biomass recalcitrance. *ACS Sustainable Chemistry & Engineering*, 5(10):8785–8795, 2017.

[6] D. A. Faux, P. J. McDonald, and N. C. Howlett. Nuclear-magnetic-resonance relaxation due to the translational diffusion of fluid confined to quasi-two-dimensional pores. *Phys. Rev. E*, 95:033116, Mar 2017.

[7] E. G. Ates, V. Domenici, M. Florek-Wojciechowska, A. Gradišek, D. Kruk, N. Maltar-Strmečki, M. Oztop, E. B. Ozvural, and A.-L. Rollet. Field-dependent NMR relaxometry for food science: Applications and perspectives. *Trends in Food Science & Technology*, 2021.

[8] P. J. Sebastião. The art of model fitting to experimental results. *European Journal of Physics*, 35(1):015017, dec 2013.

[9] P. J. Sebastião, M. J. Beira, R. Cordeiro, A. Kumar, J. C. Fernandes, A. Ferraz, and L. N. Gonçalves. The art of fitting ordinary differential equations models to experimental results. *European Journal of Physics*, 43(3):035807, mar 2022.

[10] M. Knapkiewicz, A. Rachocki, M. Bielejewski, and P. J. Sebastião. NMR studies of molecular ordering and molecular dynamics in a chiral liquid crystal with the Sm$C_\alpha{}^*$ phase. *Phys. Rev. E*, 101:052708, May 2020.

[11] M. Lehmann, S. Maisch, N. Scheuring, J. Carvalho, C. Cruz, P. J. Sebastião, and R. Y. Dong. From molecular biaxiality of real board-shaped mesogens to phase biaxiality? On the hunt for the holy grail of liquid crystal science. *Soft Matter*, 15:8496–8511, 2019.

[12] OriginLab Corporation. Origin (pro) version 2021, 2021.

[13] I. Vasilief. Qtiplot-data analysis and scientific visualisation. *Version 0.9*, 8, 2011.

[14] M. W. Maciejewski, A. W. Schuyler, M. R. Gryk, I. I. Moraru, P. R. Romero, E. L. Ulrich, H. R. Eghbalnia, M. Livny, F. Delaglio, and J. C. Hoch. NMRbox: A resource for biomolecular NMR computation. *Biophysical Journal*, 112(8):1529–1534, 2017.

[15] J.-P. Korb. Multi-scales nuclear spin relaxation of liquids in porous media. *Comptes Rendus Physique*, 11(2):192–203, 2010. Multiscale NMR and relaxation.

[16] P. Conte, L. Cinquanta, P. Lo Meo, F. Mazza, A. Micalizzi, and O. Corona. Fast field cycling NMR relaxometry as a tool to monitor parmigiano reggiano cheese ripening. *Food Research International*, 139:109845, 2021.

[17] C. Mengucci, D. Rabiti, E. Urbinati, G. Picone, R. Romano, A. Aiello, P. Ferranti, and F. Capozzi. Spotting frozen curd in PDO buffalo mozzarella cheese through insights on its supramolecular structure acquired by 1H TD-NMR relaxation experiments. *Applied Sciences*, 11(4):1466, 2021.

[18] P. Bodart, A. Rachocki, J. Tritt-Goc, B. Michalke, P. Schmitt-Kopplin, T. Karbowiak, and R. Gougeon. Quantification of manganous ions in wine by NMR relaxometry. *Talanta*, 209:120561, 2020.

[19] A. J. Simpson, M. J. Simpson, and R. Soong. Nuclear magnetic resonance spectroscopy and its key role in environmental research. *Environmental Science & Technology*, 46(21):11488–11496, 2012. PMID: 22909253.

[20] B. P. Kelleher and A. J. Simpson. Humic substances in soils: Are they really chemically distinct? *Environmental Science & Technology*, 40(15):4605–4611, 2006. PMID: 16913113.

[21] A. J. Simpson, G. Song, E. Smith, B. Lam, E. H. Novotny, and M. H. B. Hayes. Unraveling the structural components of soil humin by use of solution-state nuclear magnetic resonance spectroscopy. *Environmental Science & Technology*, 41(3):876–883, 2007. PMID: 17328197.

[22] V. Bortolotti, G. Landi, and F. Zama. 2DNMR data inversion using locally adapted multi-penalty regularization. *Computational Geosciences*, 25:1215–1228, 2021.

[23] T. Osheter, S. Campisi Pinto, C. Randieri, A. Perrotta, C. Linder, and Z. Weisman. Semi-autonomic AI LF-NMR sensor for industrial prediction of edible oil oxidation status. *Sensors*, 23(4), 2023.

[24] D. Hansen. Using deep neural networks to reconstruct non-uniformly sampled NMR spectra. *Journal of Biomolecular NMR*, 73, 11 2019.

[25] B. Halle, H. Jóhannesson, and K. Venu. Model-free analysis of stretched relaxation dispersions. *Journal of Magnetic Resonance*, 135(1):1–13, 1998.

[26] P. Lo Meo, S. Terranova, A. Di Vincenzo, D. Chillura Martino, and P. Conte. Heuristic algorithm for the analysis of fast field cycling (FFC) NMR dispersion curves. *Analytical Chemistry*, 93(24):8553–8558, 2021.

[27] H. Engl, M. Hanke, and A. Neubauer. *Regularization of Inverse Problems*. Springer Dordrecht, 2000.

[28] A. Tarantola. *Inverse problem theory and methods for model parameter estimation*. SIAM, 2005.

[29] M. W. I. Schmidt, M. S. Torn, S. Abiven, T. Dittmar, G. Guggenberger, I. A. Janssens, M. Kleber, I. Kögel-Knabner, J. Lehmann, D. A. C. Manning, P. Nannipieri, D. P. Rasse, S. Weiner, and S. E. Trumbore. Persistence of soil organic matter as an ecosystem property. *Nature*, 478:49–56, 2011.

[30] L. I. Aluwihare, D. J. Repeta, S. Pantoja, and C. J. Johnson. Two chemically distinct pools of organic nitrogen accumulate in the ocean. *Science*, 308(5724):1007–1010, 2005.

[31] M. R. Seger and G. E. Maciel. NMR investigation of the behavior of an organothiophosphate pesticide, chlorpyrifos, sorbed on soil components. *Environmental Science & Technology*, 40(3):791–796, 2006. PMID: 16509320.

[32] S. R. Mashhadi, D. Grombacher, D. Zak, P. E. Lærke, H. E. Andersen, C. C. Hoffmann, and R. J. Petersen. Borehole nuclear magnetic resonance as a promising 3D mapping tool in peatland studies. *Geoderma*, 443:116814, 2024.

[33] R. Colosimo, M. Gabriele, M. Cifelli, V. Longo, V. Domenici, and L. Pucci. The effect of sourdough fermentation on triticum dicoccum from garfagnana: 1H NMR characterization and analysis of the antioxidant activity. *Food Chemistry*, 305:125510, 2020.

[34] M. F. Marcone, S. Wang, W. Albabish, S. Nie, D. Somnarain, and A. Hill. Diverse food-based applications of nuclear magnetic resonance (NMR) technology. *Food Research International*, 51(2):729–747, 2013.

[35] P. Chinachoti, E. Vittadini, P. Chatakanonda, and Y. Vodovotz. *Characterization of Molecular Mobility in Carbohydrate Food Systems by NMR*, pages 1703–1712. Springer Netherlands, Dordrecht, 2006.

[36] L. Martí-Bonmatí and M. Kormano. MR equipment acquisition strategies: low-field or high-field scanners. *European Radiology*, 7:S263–S268, 1997.

[37] K. Ito and B. Jin. *Inverse Problems: Tikhonov Theory And Algorithms*. Series On Applied Mathematics. World Scientific Publishing Company, 2014.

[38] L. Grippo and M. Sciandrone. Globally convergent block-coordinate techniques for unconstrained optimization. *Optimization Methods and Software*, 10(4):587–637, 1999.

[39] L. Grippo and M. Sciandrone. On the convergence of the block nonlinear Gauss–Seidel method under convex constraints. *Operations Research Letters*, 26(3):127–136, 2000.

[40] V. Bortolotti, R. J. S. Brown, P. Fantazzini, G. Landi, and F. Zama. Uniform penalty inversion of two-dimensional NMR relaxation data. *Inverse Problems*, 33(1):015003, dec 2016.

[41] I. I. Rabi, J. R. Zacharias, S. Millman, and P. Kusch. A new method of measuring nuclear magnetic moment. *Phys. Rev.*, 53:318–318, Feb 1938.

[42] E. M. Purcell, H. C. Torrey, and R. V. Pound. Resonance absorption by nuclear magnetic moments in a solid. *Phys. Rev.*, 69:37–38, Jan 1946.

[43] F. Bloch, W. W. Hansen, and Martin Packard. Nuclear induction. *Phys. Rev.*, 69:127–127, Feb 1946.

[44] P. Dirac. *The Principles of Quantum Mechanics*. International series of monographs on physics. Clarendon Press, 1981.

[45] E. L. Hahn. Spin echoes. *Phys. Rev.*, 80:580–594, 11 1950.

[46] A. Abragam. *The Principles of Nuclear Magnetism*. Clarendon, Oxford, 1961.

[47] G. Whitfield and A. G. Redfield. Paramagnetic resonance detection along the polarizing field direction. *Phys. Rev.*, 106:918–920, Jun 1957.

[48] F. Noack. Nuclear magnetic relaxation spectroscopy. In *NMR*, pages 83–144, Berlin, Heidelberg, 1971. Springer Berlin Heidelberg.

[49] E. Anoardo, G. Galli, and G. Ferrante. Fast-field-cycling NMR: Applications and instrumentation. *Applied Magnetic Resonance*, 20(3):365–404, 2001.

[50] R. Kimmich. *Field-cycling NMR Relaxometry: Instrumentation, Model Theories and Applications*. The Royal Society of Chemistry, 10 2018.

[51] R. J. S. Brown. The earth's-field NML development at chevron. *Concepts in Magnetic Resonance*, 13(6):344–366, 2001.

[52] E. M. Purcell and R. V. Pound. A Nuclear Spin System at Negative Temperature. *Phys. Rev.*, 81:279–280, 1951.

[53] V. Bortolotti, L. Brizi, G. Landi, C. Testa, and F. Zama. Introduction to FFC NMR Theory and Models for Complex and Confined Fluids. In *The Environment in a Magnet: Applications of NMR Techniques to Environmental Problems*. Royal Society of Chemistry, 03 2024.

[54] R. K. Wangsness and F. Bloch. The dynamical theory of nuclear induction. *Phys. Rev.*, 89:728–739, Feb 1953.

[55] A. G. Redfield. On the theory of relaxation processes. *IBM Journal of Research and Development*, 1(1):19–31, 1957.

[56] N. Bloembergen, E. M. Purcell, and R. V. Pound. Relaxation effects in nuclear magnetic resonance absorption. *Phys. Rev.*, 73:679–712, Apr 1948.

[57] P. Levitz. Interfacial and intermittent dynamics of water in colloidal systems as probed by fast field-cycling relaxometry. In *Field-cycling NMR Relaxometry: Instrumentation, Model Theories and Applications*. The Royal Society of Chemistry, 10 2018.

[58] P.E. Levitz and J.-P. Korb. Probing glass transition of clay colloids by NMR relaxometry: Interplay between fluid brownian dynamics and particle jamming. *Europhysics Letters*, 70(5):684, apr 2005.

[59] J.-P. Korb. Nuclear magnetic relaxation of liquids in porous media. *New Journal of Physics*, 13(3):035016, 2011.

[60] D. A. Faux, S.-H. P. Cachia, P. J. McDonald, J. S. Bhatt, N. C. Howlett, and S. V. Churakov. Model for the interpretation of nuclear magnetic resonance relaxometry of hydrated porous silicate materials. *Phys. Rev. E*, 91:032311, Mar 2015.

[61] J.-P. Korb, G. Freiman, B. Nicot, and P. Ligneul. Dynamical surface affinity of diphasic liquids as a probe of wettability of multimodal porous media. *Phys. Rev. E*, 80:061601, Dec 2009.

[62] F. Barberon, J.-P. Korb, D. Petit, V. Morin, and E. Bermejo. Probing the surface area of a cement-based material by nuclear magnetic relaxation dispersion. *Phys. Rev. Lett.*, 90:116103, Mar 2003.

[63] D. A. Faux and P. J. McDonald. Explicit calculation of nuclear-magnetic-resonance relaxation rates in small pores to elucidate molecular-scale fluid dynamics. *Phys. Rev. E*, 95:033117, Mar 2017.

[64] D. A. Faux and P. J. McDonald. Nuclear-magnetic-resonance relaxation rates for fluid confined to closed, channel, or planar pores. *Phys. Rev. E*, 98(6):063110, 2018.

[65] R. Kogon and D. A. Faux. 3tm: Software for the 3-tau model. *SoftwareX*, 17:100979, 2022.

[66] G. Lipari and A. Szabo. Model-free approach to the interpretation of nuclear magnetic resonance relaxation in macromolecules. 1. Theory and range of validity. *Journal of the American Chemical Society*, 104(17):4546–4559, 1982.

[67] D. P. Bertsekas. Projected newton methods for optimization problems with simple constraints. *SIAM Journal on Control and Optimization*, 20(2):221–246, 1982.

[68] E. M. Gafni and D. P. Bertsekas. Two-metric projection methods for constrained optimization. *SIAM Journal on Control and Optimization*, 22(6):936–964, 1984.

[69] L. Grippo and M. Sciandrone. *Metodi di ottimizzazione non vincolata*. Springer Science & Business Media, 2011.

[70] M. Elad. *Sparse and Redundant Representations: From Theory to Applications in Signal and Image Processing*. Springer Publishing Company, Incorporated, 1st edition, 2010.

[71] A. N. Tikhonov. On the stability of inverse problems. *C. R. (Doklady) Acad. Sci. URSS (N.S.)*, 39:176–179, 1943.

[72] A. N. Tikhonov. On the regularization of ill-posed problems. *Dokl. Akad. Nauk SSSR*, 153:49–52, 1963.

[73] A. N. Tikhonov. On the solution of ill-posed problems and the method of regularization. *Dokl. Akad. Nauk SSSR*, 151:501–504, 1963.

[74] C. R. Vogel. *Computational methods for inverse problems*. SIAM, 2002.

[75] M. J. D. Powell. On search directions for minimization algorithms. *Mathematical Programming*, 4:193–201, 1973.

[76] D. P. Bertsekas. Nonlinear programming. *Journal of the Operational Research Society*, 48(3):334–334, 1997.

[77] P. R. Hansen. *Rank-deficient and discrete ill-posed problems*. SIAM Monographs on Mathematical Modeling and Computation. Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1998.

[78] T. Bonesky. Morozov's discrepancy principle and Tikhonov-type functionals. *Inverse Problems*, 25(1):015015, 2008.

[79] C. Clason and B. Jin. A semismooth Newton method for nonlinear parameter identification problems with impulsive noise. *SIAM Journal on Imaging Sciences*, 5(2):505–536, 2012.

[80] K. Ito, B. Jin, and T. Takeuchi. A regularization parameter for nonsmooth Tikhonov regularization. *SIAM Journal on Scientific Computing*, 33(3):1415–1438, 2011.

[81] B. Jin and J. Zou. Augmented Tikhonov regularization. *Inverse Problems*, 25(2):025001, dec 2008.

[82] T. Reginska. A regularization parameter in discrete ill-posed problems. *SIAM J. Sci. Comput.*, 17:740–749, 1996.

[83] K. Kunisch and J. Zou. Iterative choices of regularization parameters in linear inverse problems. *Inverse Problems*, 14:1247–1264, 1998.

[84] S. Lu and P. Mathé. Heuristic parameter selection based on functional minimization: Optimality and model function approach. *Math. Comput.*, 82:1609–1630, 2013.

[85] P. R. Johnston and R. M. Gulrajani. A new method for regularization parameter determination in the inverse problem of electrocardiography. *IEEE Transactions on Biomedical Engineering*, 44:19–39, 1997.

[86] P. R. Johnston and R. M. Gulrajani. An analysis of the zero-crossing method for choosing regularization parameters. *SIAM J. Sci. Comput.*, 24:428–442, 2002.

[87] B. Blümich. *Essential NMR: for Scientists and Engineers*. Springer, 2005.

[88] K. Miller. Least squares methods for ill-posed problems with a prescribed bound. *SIAM Journal on Mathematical Analysis*, 1(1):52–74, 1970.

[89] K. P. Murphy. *Machine Learning: A Probabilistic Perspective*. The MIT Press, 2012.

[90] S. Shalev-Shwartz and S. Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, USA, 2014.

[91] F. Bach. *Learning Theory from First Principles*. The MIT Press, 2024.

[92] I. J. Goodfellow, Y. Bengio, and A. Courville. *Deep Learning*. MIT Press, Cambridge, MA, USA, 2016. http://www.deeplearningbook.org.

[93] G. H. Golub, M. Heath, and G. Wahba. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 21(2):215–223, 1979.

[94] H. J. Kelley. Gradient theory of optimal flight paths. *ARS Journal*, 30:947–954, 1960.

[95] S. Dreyfus. The numerical solution of variational problems. *Journal of Mathematical Analysis and Applications*, 5(1):30–45, 1962.

[96] S. Linnainmaa. Taylor expansion of the accumulated rounding error. *BIT Numerical Mathematics*, 16(2):146–160, 1976.

[97] Y. LeCun. A theoretical framework for back-propagation. 1988.

[98] David E. R., Geoffrey E. H., and Ronald J. W. Learning representations by back-propagating errors. *Nature*, 323:533–536, October 1986.

[99] Matlab. Deep learning toolbox. https://it.mathworks.com/products/deep-learning.html.

[100] Herbert Robbins and Sutton Monro. A Stochastic Approximation Method. *The Annals of Mathematical Statistics*, 22(3):400 – 407, 1951.

[101] L. Bottou and Yann LeCun. Large scale online learning. In *Advances in Neural Information Processing Systems*, volume 16. MIT Press, 2003.

[102] D. R. Wilson and T. R. Martinez. The general inefficiency of batch training for gradient descent learning. *Neural Networks*, 16(10):1429–1451, 2003.

[103] B. T. Polyak. Some methods of speeding up the convergence of iteration methods. *USSR Computational Mathematics and Mathematical Physics*, 4(5):1–17, 1964.

[104] M. Schmidt, N. Le Roux, and F. Bach. Minimizing finite sums with the stochastic average gradient. *Math. Program.*, 162(1–2):83–112, March 2017.

[105] D. Kingma and J. Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, San Diega, CA, USA, 2015.

[106] D. H. Wolpert and W. G. Macready. No free lunch theorems for optimization. *IEEE Transactions on Evolutionary Computation*, 1(1):67–82, 1997.

[107] D. H. Wolpert. The supervised learning no-free-lunch theorems. 2002.

[108] G. Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2(4):303–314, 1989.

[109] A. R. Barron. Universal approximation bounds for superpositions of a sigmoidal function. *IEEE Transactions on Information Theory*, 39(3):930–945, 1993.

[110] H. N. Mhaskar, L. Qianli, and T. A. Poggio. When and why are deep networks better than shallow ones? In *AAAI Conference on Artificial Intelligence*, 2017.

[111] Z. Lu, H. Pu, F. Wang, Z. Hu, and L. Wang. The expressive power of neural networks: A view from the width. In *Neural Information Processing Systems*, 2017.

[112] T. A. Poggio, H. Mhaskar, L. Rosasco, B. Miranda, and Q. Liao. Why and when can deep-but not shallow-networks avoid the curse of dimensionality: A review. *International Journal of Automation and Computing*, 14(5):503–519, 2017.

[113] Y. LeCun. *Generalization and network design strategies*. Elsevier, 1989.

[114] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-8(6):679–698, 1986.

[115] D. C. Marr and E. C. Hildreth. Theory of edge detection. *Proceedings of the Royal Society of London. Series B. Biological Sciences*, 207:187 – 217, 1979.

[116] D. H. Hubel and T. N. Wiesel. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of Physiology*, 160, 1962.

[117] D. H. Hubel and T. N. Wiesel. Receptive fields of single neurones in the cat's striate cortex. *The Journal of Physiology*, 148, 1959.

[118] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[119] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on Machine Learning*, volume 37 of *Proceedings of Machine Learning Research*, pages 448–456, Lille, France, 07–09 Jul 2015. PMLR.

[120] P. Ramachandran, B. Zoph, and Q. V. Le. Searching for activation functions. *ArXiv*, abs/1710.05941, 2018.

[121] M. T. McCann, K. H. Jin, and M. Unser. Convolutional neural networks for inverse problems in imaging: A review. *IEEE Signal Processing Magazine*, 34(6):85–95, 2017.

[122] A. Mahendran and A. Vedaldi. Understanding deep image representations by inverting them, 2014.

[123] J. Johnson, A. Alahi, and L. Fei-Fei. Perceptual losses for real-time style transfer and super-resolution, 2016.

[124] Z. Zhang, Y. Song, and H. Qi. Age progression/regression by conditional adversarial autoencoder, 2017.

[125] K. Gregor and Y. LeCun. Learning fast approximations of sparse coding. In *International Conference on Machine Learning*, 2010.

[126] T. Chen, X. Chen, W. Chen, H. Heaton, J. Liu, Z. Wang, and W. Yin. Learning to optimize: A primer and a benchmark. *J. Mach. Learn. Res.*, 23:189:1–189:59, 2021.

[127] J. Adler and O. Öktem. Solving ill-posed inverse problems using iterative deep neural networks. *Inverse Problems*, 33, 2017.

[128] V. Monga, Y. Li, and Y. C. Eldar. Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing. *IEEE Signal Processing Magazine*, 38:18–44, 2019.

[129] J. Adler and O. Öktem. Learned primal-dual reconstruction. *IEEE Transactions on Medical Imaging*, 37:1322–1332, 2017.

[130] Y. Yang, J. Sun, H. Li, and Z. Xu. Admm-csnet: A deep learning approach for image compressive sensing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42:521–538, 2020.

[131] A. Sinha, P. Malo, and K. Deb. A review on bilevel optimization: From classical to evolutionary approaches and applications. *IEEE Transactions on Evolutionary Computation*, 22(2):276–295, 2018.

[132] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. C. Courville, and Y. Bengio. Generative adversarial nets. In *Neural Information Processing Systems*, 2014.

[133] D. P. Kingma and M. Welling. Auto-encoding variational bayes. *CoRR*, abs/1312.6114, 2013.

[134] S. V. Venkatakrishnan, C. A. Bouman, and B. Wohlberg. Plug-and-play priors for model-based reconstruction. *2013 IEEE Global Conference on Signal and Information Processing - GlobalSIP*, 2013.

[135] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. J. Goodfellow, and R. Fergus. Intriguing properties of neural networks. *CoRR*, abs/1312.6199, 2013.

[136] N. M. Gottschling, V. Antun, B. Adcock, and A. C. Hansen. The troublesome kernel: why deep learning for inverse problems is typically unstable. *ArXiv*, abs/2001.01258, 2020.

[137] S. Boyd, N. Parikh, E. Chu, B. Peleato, and J. Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Found. Trends Mach. Learn.*, 3(1):1–122, January 2011.

[138] M. V. Afonso, J. M. Bioucas-Dias, and M. A. T. Figueiredo. Fast image recovery using variable splitting and constrained optimization. *IEEE Transactions on Image Processing*, 19(9):2345–2356, 2010.

[139] P. L. Combettes and J.-C. Pesquet. *Proximal Splitting Methods in Signal Processing*, pages 185–212. Springer New York, New York, NY, 2011.

[140] N. Parikh and S. Boyd. Proximal algorithms. *Found. Trends Optim.*, 1(3):127–239, January 2014.

[141] J. J. Moreau. Proximité et dualité dans un espace hilbertien. *Bulletin de la Société Mathématique de France*, 93:273–299, 1965.

[142] S. Sreehari, S. Venkatakrishnan, B. Wohlberg, L. F. Drummy, J. P. Simmons, and C. A. Bouman. Plug-and-play priors for bright field electron tomography and sparse interpolation. *IEEE Transactions on Computational Imaging*, 2:408–423, 2015.

[143] S. H. Chan, X. Wang, and O. A. Elgendy. Plug-and-play admm for image restoration: Fixed-point convergence and applications. *IEEE Transactions on Computational Imaging*, 3:84–98, 2016.

[144] T. Meinhardt, M. Möller, C. Hazirbas, and D. Cremers. Learning proximal operators: Using denoising networks for regularizing inverse imaging problems. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 1799–1808, 2017.

[145] G. Mataev, M. Elad, and P. Milanfar. Deepred: Deep image prior powered by red. *ArXiv*, abs/1903.10176, 2019.

[146] E. T. Reehorst and P. Schniter. Regularization by denoising: Clarifications and new interpretations. *IEEE Transactions on Computational Imaging*, 5:52–67, 2018.

[147] A. M. Teodoro, K. M. Bioucas-Dias, and M. A. T. Figueiredo. A convergent image fusion algorithm using scene-adapted gaussian-mixture-based denoising. *Trans. Img. Proc.*, 28(1):451–463, January 2019.

[148] A. Kazerouni, U. S. Kamilov, E. Bostan, and M. Unser. Bayesian denoising: From MAP to MMSE using consistent cycle spinning. *IEEE Signal Processing Letters*, 20(3):249–252, 2013.

[149] R. Gribonval. Should penalized least squares regression be interpreted as maximum a posteriori estimation? *IEEE Transactions on Signal Processing*, 59(5):2405–2410, 2011.

[150] C. Park, S. Shoushtari, W. Gan, and U. S. Kamilov. Convergence of nonconvex PnP-ADMM with MMSE denoisers. *2023 IEEE 9th International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, pages 511–515, 2023.

[151] S. Hurault, A. Leclaire, and N. Papadakis. Gradient step denoiser for convergent plug-and-play. *ArXiv*, abs/2110.03220, 2021.

[152] S. Hurault, A. Chambolle, A. Leclaire, and N. Papadakis. A relaxed proximal gradient descent algorithm for convergent plug-and-play with proximal denoiser. In *Scale Space and Variational Methods in Computer Vision: 9th International Conference, SSVM 2023, Santa Margherita Di Pula, Italy, May 21–25, 2023, Proceedings*, page 379–392, Berlin, Heidelberg, 2023. Springer-Verlag.

[153] R. Gribonval and M. Nikolova. A characterization of proximity operators. *J. Math. Imaging Vis.*, 62(6–7):773–789, July 2020.

[154] V. Bortolotti, P. Conte, G. Landi, P. Lo Meo, A. Nagmutdinova, G. V. Spinelli, and F. Zama. Robust algorithms for the analysis of fast-field-cycling nuclear magnetic resonance dispersion curves. *Computers*, 13(6), 2024.

[155] G. Landi, G. V. Spinelli, F. Zama, D. Chillura Martino, P. Conte, P. Lo Meo, and V. Bortolotti. An automatic L1-based regularization method for the analysis of FFC dispersion profiles with quadrupolar peaks. *Applied Mathematics and Computation*, 444:127809, 2023.

[156] P. C. Hansen. Truncated singular value decomposition solutions to discrete ill-posed problems with ill-determined numerical rank. *SIAM Journal on Scientific and Statistical Computing*, 11(3):503–518, 1990.

[157] G. H. Golub and C. F. Van Loan. *Matrix computations*. JHU press, 2013.

[158] S.-J. Kim, K. Koh, M. Lustig, S. P. Boyd, and D. M. Gorinevsky. An interior-point method for large-scale ℓ_1-regularized least squares. *IEEE Journal of Selected Topics in Signal Processing*, 1:606–617, 2007.

[159] A. Beck and M. Teboulle. A Fast Iterative Shrinkage-Thresholding Algorithm for linear inverse problems. *SIAM Journal on Imaging Sciences*, 2(1):183–202, 2009.

[160] D. A. Faux, O. Istók, A. A. Rahaman, P. J. McDonald, E. McKiernan, and D. F. Brougham. Nuclear spin relaxation in aqueous paramagnetic ion solutions. *Physical Review E*, 107(5):054605, 2023.

[161] J. Kowalewski, D. Kruk, G. Parigi, et al. NMR relaxation in solution of paramagnetic complexes: Recent theoretical progress for s¿= 1. *Advances in inorganic chemistry*, 57:41–104, 2005.

[162] P. Conte and N. Nestle. Water dynamics in different biochar fractions. *Magnetic Resonance in Chemistry*, 53(9):726–734, 2015.

[163] C. De Pasquale, V. Marsala, A. E Berns, M. Valagussa, A. Pozzi, G. Alonzo, and P. Conte. Fast field cycling NMR relaxometry characterization of biochars obtained from an industrial thermochemical process. *Journal of soils and sediments*, 12:1211–1221, 2012.

[164] R. C. Smith. *Uncertainty quantification: theory, implementation, and applications*. SIAM, 2013.

[165] D. Kruk, P. Rochowski, M. Florek-Wojciechowska, P. J. Sebastião, D. J. Lurie, and L. M. Broche. ^1H spin-lattice NMR relaxation in the presence of residual dipolar interactions–dipolar relaxation enhancement. *Journal of Magnetic Resonance*, 318:106783, 2020.

[166] J. Nocedal and S. J. Wright. *Numerical Optimization*. Springer, New York, NY, USA, 2e edition, 2006.

[167] G. V. Spinelli, D. Evangelista, L. Hu, and F. Zama. Neural network-based inversion of NMR dispersion profiles for enhanced analysis of food systems. *Neural Computing and Applications*, 2024.

[168] G. Landi, G. V. Spinelli, F. Zama, D. Martino Chillura, P. Conte, P. Lo Meo, and V. Bortolotti. Modelfreeffc. `https://site.unibo.it/softwaredicam/en/modelfree`, 2021. Freeware standalone Matlab® tool for fitting nuclear magnetic relaxation dispersion (NMRD) curves obtained by fast field cycling nuclear magnetic resonance (FFC-NMR) relaxometry measurements.

[169] U. S. Kamilov, C. A. Bouman, G. T. Buzzard, and B Wohlberg. Plug-and-play methods for integrating physical and learned models in computational imaging. *IEEE Signal Processing Magazine*, 40(1):85 – 97, 2023.