Compressed Sensing and Experiment Design in Magnetic Field Based Imaging

Vom Promotionsausschuss der
Technischen Universität Hamburg

zur Erlangung des akademischen Grades

Doktor der Naturwissenschaften (Dr. rer. nat)

genehmigte Dissertation

von
Mirco Grosser

aus
Hamburg

2023

Authors's ORCID: https://orcid.org/0000-0001-8005-8859
DOI: https://doi.org/10.15480/882.8925

1. Gutachter: Prof. Dr. Tobias Knopp
2. Gutachter: Prof. Dr. Alexander Penn
Tag der mündlichen Prüfung: 24.10.2023

Abstract

Magnetic-field-based imaging is an important class of imaging modalities that allow anatomical and functional imaging with a high resolution and without the potentially harmful use of ionizing radiation. This class includes magnetic resonance imaging (MRI), which is popular in clinical applications due to its excellent soft-tissue contrast and high spatial resolution. Additionally, MRI provides not only anatomical information but can also yield a variety of functional information. A second modality is magnetic particle imaging (MPI), which allows imaging the distribution of superparamagnetic iron oxide nanoparticles. This modality allows highly sensitive functional imaging with high spatial resolution and frame rates of up to 50 Hz. MRI and MPI not only have similarities in the underlying physics but they also complement each other well in terms of their imaging properties and the types of information that they provide. For instance, MPI allows imaging of vascular processes at times scales not reachable in MRI, whereas MRI can provide the anatomical information needed for the interpretation of MPI images.

In both MRI and MPI measurement times are a limiting factor, even though the bottlenecks occur in different places of the imaging pipeline. In MRI, the data acquisition itself is known to be a bottleneck leading to comparably long measurement times. For MPI, a similar limitation arises from the fact that image reconstruction relies on a so-called system matrix to encode the imaging process. This system matrix is most commonly acquired in a time-consuming calibration scan. Mathematically, both of these problems share many similarities and can be tackled using the same kinds of methods from the field of compressed sensing (CS). The latter makes use of modern sampling methods to reduce the number of measurements performed. The missing information is than recovered using sophisticated mathematical optimization methods, which exploit prior knowledge about the structure of the underlying signals. This work investigates the application of CS in the fields of MRI and MPI and aims to address some of the challenges arising in both modalities.

The emergence of compressed sensing reflects a general trend in imaging, where more and more sophisticated algorithms are used to enhance image quality and mitigate problems that cannot easily be addressed by improving the scanner hardware. A key requirement for the development and application of such algorithms are software frameworks that are both accessible and highly performant. To address this need, this work describes an optimization framework written in the programming language Julia, which can serve as the basis for image reconstruction frameworks for multiple imaging modalities. Based on this, dedicated image reconstruction frameworks were developed for the modalities MRI and MPI. Both frameworks provide a high degree of accessibility. In contrast to most other frameworks, they are implemented in a single programming language, which makes it easy to customize and extend them. In a

benchmark, it is shown that the performance of our framework is indeed on par with other state-of-the-art image reconstruction frameworks.

A particularly promising technique to accelerate MRI acquisitions is the use of non-Cartesian trajectories with long readouts. A drawback associated with these trajectories is their sensitivity to inhomogeneities of the main magnetic field, which can lead to image artifacts. These artifacts can be mitigated during reconstruction by employing a B_0-aware signal model that incorporates information about the present field inhomogeneities. As a matter of fact, this approach greatly increases reconstruction time, especially when combining it with CS-type reconstruction schemes. To address this issue, this work investigates the acceleration of B_0-aware image reconstruction on multiple levels. We propose a new method to approximate the B_0-aware imaging operator based on the randomized singular value decomposition. This approach efficiently reduces the number of computations to be performed during image reconstruction. We combine this with an efficient preconditioner, which is applied in k-space, i.e. the domain of MRI measurements, and which significantly reduces the number of iterations required for image reconstruction.

Whereas CS-based image reconstruction has been heavily researched for MRI, only few works have investigated the CS-based recovery of MPI system matrices. Moreover, all of these works have focused on the same form of prior knowledge based on the sparsity of MPI system matrices in the domain of the discrete cosine transform. In this work, we aim to enhance system matrix recovery by incorporating further forms of prior knowledge. More precisely, we show that the rows of the MPI system matrix can be modeled as low-rank tensors. Based on this, we propose a new algorithm which exploits both the sparsity of the system matrix and the proposed low-rank structure. Our experiments show that this approach leads to an improved quality of the recovered system matrices.

The last part of this thesis, turns to the question how to optimally choose the set of measurements to be acquired such that the expected reconstruction error is minimized. Based on the field of optimized experiment design, this work proposes a framework that adapts the set of measurements to the expected sparsity structure of the signal class at hand. The proposed method makes use of the Bayesian Fisher information matrix, which has several advantages over other frequentist approaches. On one hand, the used prior provides us with an additional mechanism to incorporate prior information about the given signal class. On the other hand, most experiment design approaches suffer from a high numerical complexity, which may prevent their application to large datasets. Adopting a Bayesian framework, allows us to reduce computation times and thus enables its application in a wider range of settings. To validate the method, we first apply it for the design of MRI sampling patterns. In a second step, we investigate how the method can be used to generate optimized sampling schemes for the MPI system matrix. An interesting aspect about the obtained sampling patterns is that they display a variable sampling density, which stands in contrast to the patterns that have previously been used for the sampling of MPI system matrices.

Kurzfassung

Die Magnetfeld-basierte Bildgebung bezeichnet eine Klasse an Bildgebungsmodalitäten, welche eine anatomische und funktionelle Bildgebung mit hoher Auflösung, und ohne den schädlichen Einsatz ionisierender Strahlung, ermöglicht. Zu dieser Klasse gehört die Magnetresonanztomografie (MRT), die in der klinischen Anwendung aufgrund ihres guten Weichteilkontrasts und der hohen räumlichen Auflösung beliebt ist. Darüber hinaus liefert die MRT nicht nur anatomische Informationen, sondern kann auch eine Vielzahl funktioneller Bildgebungsinformationen generieren. Eine weitere Modalität ist die Magnetpartikelbildgebung (MPI) mit der sich die räumliche Verteilung von superparamagnetischen Eisenoxid-Nanopartikeln bestimmen lässt. Dieses Verfahren ermöglicht eine hochempfindliche, funktionelle Bildgebung mit hoher räumlicher Auflösung und Bildraten von bis zu 50 Hz. MRT und MPI haben nicht nur Ähnlichkeiten in der zugrunde liegenden Physik, sondern sie ergänzen sich auch hinsichtlich der Art der Informationen, die sie liefern. Beispielsweise ermöglicht MPI die Darstellung vaskulärer Prozesse mit einer Auflösung, welche bei der MRT nicht erreicht werden kann. Andererseits kann die MRT die, für die Interpretation von MPI-Bildern notwendigen, anatomischen Informationen liefern.

Sowohl in der MRT als auch beim MPI stellen lange Messzeiten eine wichtige Herausforderung dar, auch wenn die Limitierungen an verschiedenen Stellen auftreten. Eine bekannte Limitierung in der MRT sind die langen Zeiten, welche für die Aufnahme der Bildgebungsdaten benötigt werden. In der Magnetpartikelbildgebung entsteht eine ähnliche Limitierung dadurch, dass die Bildrekonstruktion von einer sogenannten Systemmatrix abhängt, welche die Bildgebungsphysik beschreibt. Diese Systemmatrix wird üblicherweise durch eine zeitaufwendige Kalibrierungsmessung bestimmt. Aus mathematischer Sicht haben beide Probleme eine sehr ähnliche Struktur und können mithilfe der gleichen Ansätze aus dem Bereich des Compressed Sensing (CS) gelöst werden. Das Compressed Sensing ist ein relativ junges Verfahren, welches moderne Samplingtheoreme benutzt um die Zahl durchzuführender Messungen zu reduzieren. Die fehlenden Informationen werden anschließend mit Hilfe von Optimierungsmethoden, und unter Ausnutzung von Vorwissen über die erwarteten Signale, rekonstruiert. Diese Arbeit untersucht die Anwendung von CS in den Modalitäten MRT und MPI und entwickelt Lösungsansätze für einige der, in beiden Modalitäten auftretenden, Herausforderungen.

Die Entwicklung des Compressed Sensing spiegelt einen allgemeinen Trend in der Bildgebung wider, bei welchem fortschrittliche Algorithmen verwendet werden, um die Qualität der Bildgebung zu erhöhen und Probleme anzugehen, welche hardwareseitig schwer zu lösen sind. Eine Grundvoraussetzung für die Entwicklung und Anwendung solcher Algorithmen sind Software-Frameworks, welche sowohl zugänglich als auch performant sind. Aus diesem Grund wird in dieser Arbeit ein Optimierungs-Framework in der Programmiersprache Julia vorgestellt. Darauf aufbauend werden dedizierte Bildrekonstruktions-

Frameworks für die Modalitäten MRT und MPI beschrieben. Beide Frameworks bieten ein hohes Maß an Zugänglichkeit. Im Unterschied zu vielen anderen Frameworks sind sie in einer einzigen Programmiersprache implementiert, was es erleichtert sie zu erweitern und Anpassungen vorzunehmen. Die Ergebnisse des anschließend durchgeführten Benchmarks zeigen, dass die Performance der entwickelten Frameworks auf dem gleichen Niveau mit anderen State-Of-The-Art Frameworks ist.

Ein vielversprechender Ansatz zur Beschleunigung von MRT-Messungen ist die Verwendung nicht-Kartesischer Trajektorien mit langen Auslesezeiten. Ein Nachteil dieser Trajektorien ist jedoch ihre Sensitivität in Bezug auf Inhomogenitäten des Hauptmagnetfeldes, was zur Entstehung von Bildartefakten führen kann. Diese Artefakte sind vermeidbar durch die Verwendung eines erweiterten Signalmodels, welches Informationen über die gegebenen Feldinhomogenitäten beinhaltet. Leider führt dies zu einem signifikant höherem numerischen Aufwand für die Bildrekonstruktion, insbesondere wenn CS für die Rekonstruktion verwendet wird. In dieser Arbeit werden verschiedene Ansätze untersucht, um die Bildrekonstruktion unter Berücksichtigung von Feldinhomogenitäten zu beschleunigen. Auf der Ebene des Signalmodels wird ein neuer Ansatz zur Auswertung des Bildgebungsoperators, auf Grundlage der randomisierten Singulärwertzerlegung, vorgeschlagen. Dieser minimiert effektiv die Anzahl an Operationen, die für die Bildrekonstruktion nötig sind. Darüber hinaus wird der Ansatz mit einer effizienten Vorkonditionierung kombiniert, welche im k-Raum, also im Raum der MRT-Messungen, angewendet wird. Diese führt zu einer signifikanten Reduktion der, für die Bildrekonstruktion nötigen, Zahl an Iterationen.

Während die CS-basierte Bildrekonstruktion für die MRT ausführlich erforscht wurde, gibt es nur eine kleine Anzahl an Arbeiten zur CS-basierten Bestimmung von MPI Systemmatrizen. Darüber hinaus verwenden alle diese Arbeiten die gleiche Form von Vorwissen über die Dünnbesetztheit der Systemmatrix im Definitionsbereich der diskreten Kosinustransformation. In dieser Arbeit wird die Bestimmung von Systemmatrizen verbessert, indem weitere Formen von Vorwissen ausgenutzt werden. Hierzu wird zuerst gezeigt, dass die Zeilen der MPI Systemmatrix als Tensoren mit niedrigem Rang modelliert werden können. Darauf aufbauend wird ein Algorithmus vorgeschlagen, welcher sowohl die oben erwähnte Dünnbesetztheit als auch die vorgestellte Tensor-Darstellung ausnutzt. Die durchgeführten Experimente zeigen, dass dieser Ansatz zu einer Erhöhung der Qualität der rekonstruierten Systemmatrizen führt.

Der letzte Teil dieser Arbeit beschäftigt sich mit der Frage, wie die durchzuführenden Messungen ausgewählt werden sollten, damit der zu erwartende Rekonstruktionsfehler minimiert wird. Aufbauend auf Techniken aus dem Feld der optimalen Versuchsplanung, wird ein Framework vorgeschlagen, welches die durchzuführenden Messungen an die Dünnbesetztheits-Struktur der gegebenen Signalklasse anpasst. Dazu verwendet die entwickelte Methode die Bayessche Fisher Informationsmatrix, was mit einigen Vorteilen gegenüber anderen frequentistischen Ansätzen einhergeht. Ein Vorteil ist, dass die Verwendung einer A-Priori-Verteilung uns einen weiteren Mechanismus liefert, um Vorwissen über die wiederherzustellenden Signal in die Optimierung einfließen zu lassen. Darüber hinaus haben viele Methoden zur optimalen Versuchsplanung eine hohe numerische Komplexität, was ihre Anwendung für große Datensätze erschwert. Die Verwendung der Bayesschen Fisher Informationsmatrix ermöglicht es uns die Rechenzeit zu reduzieren und damit die Menge möglicher Anwendungen zu erhöhen. Zur Validierung

der Methode, verwenden wir sie für die Optimierung von Abtastschemata für die MRT. In einem zweiten Schritt untersuchen wir, wie die Methode zur Generierung von Messschemata für MPI Systemmatrizen verwendet werden kann. Ein interessanter Aspekt unserer Ergebnisse ist, dass die optimierten Schemata eine variable Dichte an Messpunkten aufweisen. Dies steht in Kontrast zu den bisher für die Messung von Systemmatrizen verwendeten Messschemata.

Acknowledgments

This work would have not been conceivable without the help of a large number of people, which have accompanied and supported me during this long journey throughout the last years.

First and foremost, I would like to express my gratitude to Prof. Tobias Knopp for giving me the opportunity to be part of the exciting research field of biomedical imaging. He always supported this work in numerous ways including giving advice, educating me in various aspects of imaging, mathematics and programming and proof-reading manuscripts. Finally, he created a friendly and team-oriented environment which makes research a fun endeavor.

Additionally, I would like to thank Martin Möddel for always sharing his expertise with me and teaching me a lot about scientific writing and how to best structure research projects.

During my PhD-research, I had the pleasure of working together with a variety of great colleagues, all of which contributed strongly to this work and to the amazing research environment that I was surrounded by. Therefore, my thanks go to Marija Boberg for her support with visualizations (among countless other things) and for ensuring a continuous supply with cake. Patryk Szwargulski, Florian Griese, Florian Thieben and Fabian Mohn were always ready to discuss various research ideas and they supported me immensely during my rare excursion to the MPI and MRI scanners. Niklas Hackelberg proved to be a never ending source of information on software development and his passion for it also fueled my own passion for scientific software. I am also grateful to Annemarie Tauche for handling all kinds of bureaucratic issues and organizing memorable team events. Finally, I would like to thank all the other researchers at the institute for biomedical imaging. All of them strongly impacted my research experience and my personal development during this time and I will forever remember our numerous vivid discussions during coffee breaks and lunch.

I owe a dept of gratitude to all the eyes that were proofreading this manuscript: Konrad Scheffler, Fynn Förger, Lina Nawwas, Fabian Mohn, Marija Boberg, Florian Thieben and Niklas Hackelberg. Their comments and suggestions were invaluable to me as the manuscript would not be in the present form without them.

Last but not least, I would like to thank my family for always supporting me and enabling my studies. Most importantly, I would like to thank Jasmin Klenke for her constant support in both good and bad times.

Contents

Acronyms

Acronym	Description
ADC	analog-to-digital converter
ADMM	alternating direction of multipliers method
CG	conjugate gradient
CPU	central processing unit
CRB	Cramer-Rao bound
CS	compressed sensing
CSFR	compressed sensing with fixed rank
CSLR	compressed sensing with low rank
CT	computed tomography
DCT	discrete cosine transform
DF-FOV	drive field field of view
DFT	discrete Fourier transform
FIM	Fisher information matrix
FISTA	fast iterative shrinkage-thresholding algorithm
FFP	field free point
FFT	fast Fourier transform
FOV	field of view
FS	frequency segmentation
GPU	graphics processing unit
GRE	gradient echo
HOSVD	higher order singular value decomposition
MAP	maximum a posteriori
ML	maximum likelihood
MPI	magnetic particle imaging
MRI	magnetic resonance imaging
NFFT	non-equispaced fast Fourier transform
NRMSD	normalized root mean squared deviation
NSR	noise-to-signal ratio
NSST	Nyquist-Shannon sampling theorem

Acronym	Description
PD	Poisson disk
PDHG	primal dual hybrid gradient
PET	positron emission tomography
PI	parallel imaging
PSF	point spread function
RF	radio frequency
SENSE	sensitivity encoding
SNR	signal-to-noise ratio
SPECT	single photon emission computed tomography
SPIO	superparamagnetic iron oxide nanoparticle
SSIM	structural similarity index
SVD	singular value decomposition
TS	time segmentation

Physical Quantities

Symbol	Description	Unit
$\mathbf{B}$	magnetic flux density	T
c	concentration	$1/\mathrm{m}^3$
$\mathbf{D}$	electric displacement field	Cm
$\mathbf{d}$	electric dipole moment	C/m^2
E	energy	J
e	elementary charge	C
$\mathbf{E}$	electric field	V/m
f	frequency	$1/\mathrm{s}$
G	gradient field strength	$\mathrm{T}/(\mu_0\mathrm{m})$
$\mathbf{H}$	magnetic field	A/m
$\hbar$	reduced Planck constant	Js
I	electric current	A
$\mathbf{j}$	electric current density	A/m^2
K	anisotropy energy density	J/m^3
k_B	Boltzmann constant	J/K
$\mathbf{L}$	angular momentum	Nms
$\mathbf{M}$	magnetization	A/m
m	mass of a particle	kg
$\mathbf{P}$	polarization	C/m^2
$\mathbf{p}$	coil sensitivity	$1/\mathrm{m}$
q	electric charge	C
$\mathbf{r}$	location of a point in space	m
S	surface of an object	m^2
T	temperature	K
t	point of time	s
T_1	longitudinal relaxation time	s
T_2	transverse relaxation time	s
T_2^*	effective transverse relaxation time	s
u	electric voltage	V

Symbol	Description	Unit
V	volume of interest	m^3
$\mathbf{v}$	velocity	$\mathrm{m/s}$
γ	gyromagnetic ratio	$\mathrm{rad/(sT)}$
ε_0	vacuum permitivity	$\mathrm{F/m}$
η	viscosity	$\mathrm{kg/ms}$
$\boldsymbol{\mu}$	magnetic dipole moment	Am^2
μ_0	vacuum permeability	$\mathrm{H/m}$
ρ	charge density	$\mathrm{C/m}^3$
$\boldsymbol{\tau}$	torque	Nm
τ_N	Néel relaxation time	s
τ_B	Brownian relaxation time	s
ω	angular frequency	rad/s

1 Introduction

1.1 Imaging Background

Imaging techniques play a fundamental role in modern medicine as they provide a non-invasive way to obtain information about the internals of the human body. As a consequence, diagnoses rely to a large extend on images obtained using a range of imaging modalities, which differ in their underlying physical principles and the achievable image contrasts. In the clinical practice, the most commonly used imaging modalities include computed tomography (CT) [1], sonography [2] and magnetic resonance imaging (MRI) [3]. All of these methods can be characterized as morphological imaging methods, because the respective image signal is directly coupled to physical parameters of the body being imaged. For instance, CT is sensitive to the x-ray attenuation coefficient of the imaged tissues, whereas the MRI signal is generated by hydrogen protons inside the body. A complementary class of imaging methods are the indirect imaging modalities. The latter use some form of indicator substance, also called tracer, which generates the detected image signal. A popular examples of these techniques is positron emission tomography (PET) where a tracer containing β^+-emitters is injected into the body [2]. In the human body the emitted positrons recombine with the present electrons, which leads to the emission of γ-rays that can be detected and used to form an image of the tracer distribution. In clinical applications, the radioactive tracer is often attached to nutrients such as glucose. Imaging of the resulting tracer distribution then gives information about metabolic activities inside the imaged patient.

All of the different imaging modalities have different strengths and weaknesses, which determine their application in the clinical practice. To provide an overview some characteristics of the modalities discussed above are summarized in Table 1.1. For example, the strengths of CT include the high achievable spatial and temporal resolution. Moreover, the images are directly related to the electron density thus allowing quantitative imaging. Its main drawback is that the patient and the medical staff are exposed to a significant amount of radiation which increase the risk of cancer. PET imaging allows quantitative imaging with a high sensitivity and it provides functional information, which is not accessible using techniques such as MRI or CT. As a matter of fact, it also relies an ionizing radiation and it only provides moderate spatial resolution.

In contrast to the previous examples, MRI is able to generate high-resolution images without the use of ionization radiation. Furthermore, it can generate a great variety of contrasts allowing both anatomical

	CT	MRI	PET	MPI
spatial resolution	0.5 mm	1 mm	4 mm	0.5 mm
measurement time	1 s	<1 s-30 min	1 min	<0.1 s
sensitivity	low	low	high	high
quantitativity	yes	no	yes	yes
risks	x-rays	heating	β/γ-radiation	heating

Table 1.1: Comparison of the key characteristics of the imaging modalities CT, MRI, PET and MPI.

imaging with excellent soft-tissue contrast as well as functional imaging of the patient at hand. As a consequence, MRI is routinely used not only for anatomical imaging, but also for imaging of blood flow [4], diffusion [5], blood oxygenation [6, 7] and other tissue properties such as their magnetic susceptibility [8]. The main limitations of MRI are its comparably low sensitivity and the long acquisition times required to obtain high-resolution images with a good signal-to-noise ratio (SNR). This imposes limits on the achievable temporal resolution and on the feasibility of certain kinds of measurements. Moreover, it should be noted that MRI contrast-agents are not as functionalized as those used in PET and that they typically lead to a negative image contrast.

An imaging modality, which tries to complement the existing selection of established modalities is magnetic particle imaging (MPI) [9, 10]. It was invented in 2005 and is currently in a preclinical stage. MPI images the distribution of superparamagnetic iron oxide nanoparticles (SPIOs), which are injected in the body prior to imaging. Importantly, it has been demonstrated that this imaging can be performed with high spatial resolution, high temporal resolution, as well as high sensitivity. Moreover, MPI promises the ability to perform quantitative imaging due to the linear relationship between particle concentration and measured signal. Due to the functional information it provides, one vision for MPI is the development of tailored molecular tracers, which enable a similar kind of physiological imaging as provided by PET, albeit without exposure to ionizing radiation. Currently investigated applications of MPI include blood flow imaging for the diagnosis of vascular diseases [11–13], the tracking of medical instruments during interventions [14, 15] and stroke imaging [16]. One of the main challenges for MPI arises due to the physically complicated relationship between tracer concentration and the associated measurement signal. In particular, the system matrix, describing this relationship, is typically acquired using a time-consuming calibration scan. The time needed for this kind of calibration often becomes a limiting factor for practical applications.

Even though both are independent imaging modalities, it appears quite natural to combine MRI and MPI in imaging experiments. The primary reason for this is that both modalities provide highly complementary information about the object being imaged. In fact, the anatomical information provided by a modality such as MRI is a prerequisite for the interpretation of the functional images provided by MPI. Beyond this, MRI is a natural candidate to complement MPI because both modalities do not rely on ionizing radiation. Finally, we note that the SPIOs used in MPI, also generate a measurable contrast in MRI images. This is interesting for joint MRI/MPI imaging, because it might allow the registration of

both datasets without the use of specialized bi-modal markers or of a dedicated MPI/MRI system that is capable of bimodal imaging [17].

Interestingly, both MRI and MPI both face very similar challenges in the sense that the acquisition of either imaging data or the system matrix is a very time consuming process. Both of these problems can be addressed by the use of modern sampling methods, which use prior knowledge about the signal to be determined in order to reduce measurement time.

1.2 Compressed Sensing & Experiment Design

The objective of this work is the development of algorithmic tools, which allow to perform high-quality image reconstruction or system calibration even when only limited data is acquired by the imaging system. In view of aforementioned problems, such tools are highly desirable, because they allow reducing the amount of data to be collected. In turn this often leads to a corresponding reduction in the time required for an imaging experiment or system calibration.

A common tool to guide the design of imaging experiments is the Nyquist-Shannon sampling theorem (NSST) [18]. It states that a signal can be recovered exactly, given that it is uniformly sampled with a frequency at least twice as large as the maximum frequency contained in the signal itself. Since the NSST holds for all band-limited signals it is very widely applicable explaining its popularity.

Compressed sensing (CS) is a relatively young sampling paradigm, which aims to improve upon classical sampling theorems such as the NSST [19, 20]. It establishes that signals can be recovered from a significantly smaller number of measurements, given that they have a sparse representation in a known domain. This makes CS an ideal starting point to achieve the aforementioned goals of reducing measurement time. Having established the sparsity of a signal, the latter can be recovered from a suitable set of measurements, by solving a non-linear inverse problem. The number of measurements required for a successful recovery is primarily determined by the sparsity of the signal at hand and the incoherence between the measurement basis and the sparsity basis [21].

Since its introduction, CS has been adopted for many applications including photography [22], aperture synthesis astronomy [23], network tomography [24] and even theoretical studies of open quantum systems [25]. In the field of biomedical imaging, CS has been applied very effectively for reducing the number of measurements needed for MRI scans [26, 27] and for reducing the number of measurements required for the measurement of the MPI system matrix [28, 29]. Despite its apparent success there remain a number of challenges, which we aim to address in this work.

1. CS requires prior knowledge about a sparse representation of the signal at hand. Finding such representations is a field of active research.
2. Signal recovery requires solving a non-linear inverse problem. This task can become quite challenging, especially for more complicated measurement models. This numerical effort can be a problem for clinical applications, e.g. in MRI.

3. While providing guarantees for the recovery of sparse signals, CS does not answer the question how to best design the set of measurements to be performed.

Concerning the first point, we note that in the field of MRI, a large number of image representations have been found and used in conjunction with CS [26, 30–32]. In the field of MPI the situation is different. Previous approaches for the recovery of MPI system matrices have all relied on a quite generic sparse representation of system matrices in the frequency domain [28]. Thus, further improvements are to be expected when more prior knowledge about the structure of MPI system matrices is exploited.

The second point is particularly relevant for clinical imaging, where it is common practice to obtain a series of datasets with different contrasts. At the same time, one aims to maximize the duty cycle of the imaging device. This naturally leads to an upper limit for the image reconstruction time that can be tolerated. In MRI, this quickly leads to problems when combining CS with non-Cartesian sampling schemes that require a more complicated signal model to avoid artifacts. On the other hand, non-Cartesian sampling schemes provide lots of potential to accelerate MRI acquisitions. For this reason, one part of this thesis investigates methods to accelerate the CS-based reconstruction of non-Cartesian MRI data.

The last point is complementary to the first two points in the sense that it is not directly concerned with the actual recovery of a given signal but with the question how an imaging experiment should be designed in order to minimize the error of the recovered signals. This task can be tackled using tools from the field of experiment design. As a matter of fact, such methods normally have a high numerical complexity and become infeasible for large-scale applications in imaging. So far, a number of works have investigated optimal sampling schemes in MRI [33–36]. In contrast, no one has yet investigated optimal sampling schemes for MPI system matrices, to the best of our knowledge.

1.3 Contributions & Publications

This work investigates the application of CS-based sampling approaches for the magnetic field based imaging modalities MRI and MPI. To put this into perspective, we note that there currently is significantly more research activity in the established field of MRI than in the young field of MPI. Hence, one aspect of this work is to enable some transfer of methods between the fields. While such a transfer can be very beneficial, it is also challenging because of the differences in the way that image reconstruction is commonly performed in both modalities. To facilitate this transfer, this work focuses on a generic mathematical framework and the development of software tools to facilitate the development of algorithms across imaging modalities. Thematically, two major aspects are covered. One part of this work is concerned with the development of algorithms for image reconstruction based on the given measurement data. Notably, this leads to the questions what kinds of prior knowledge to incorporate in an image reconstruction method and how to design algorithms that are efficient in terms of their runtime and computational complexity. The second part of this work is concerned with the problem of finding optimized sampling schemes such that the expected reconstruction error for a given class of signals becomes minimal.

In particular, this work encompasses the following contributions:

- A basic requirement for the development and application of image reconstruction algorithms is a suitable software framework, which provides the required optimization tools and which is both performant and easily extendable. Currently, most software frameworks use a hybrid approach based on two different programming languages to fulfill these requirements [37–40]. This approach is suboptimal because it often limits the flexibility and accessibility of the software framework. To provide an alternative, this work presents a new and flexible optimization framework, which is implemented purely in the programming language Julia [41]. The implemented functionality constitutes a central building block of the MRI reconstruction framework *MRIReco.jl*, which I developed together with Tobias Knopp. Similarly, it underlies the MPI reconstruction framework *MPIReco.jl*, which was developed by Tobias Knopp in cooperation with Martin Möddel, Patryk Szwargulski, Florian Griese, Marija Boberg and myself. In particular, both frameworks combine high computational performance with a flexible and easily-extendable interface, thus facilitating the development of new algorithms.
- In MRI, the reconstruction of undersampled, non-Cartesian data poses a significant numerical challenge. On one hand, these acquisitions often require a B_0-aware signal model, which is hard to evaluate numerically [42]. On the other hand, commonly used optimization methods suffer from slow convergence caused by the non-Cartesian sampling [43, 44]. To address these issues, this work proposes an efficient approximation to the B_0-aware signal model, which is based on the randomized singular value decomposition (SVD). To accelerate convergence, we investigate the use of diagonal k-space preconditioners and provide an algorithm to compute them for the B_0-aware signal model.
- While many different signal representations have been found and used for MRI image reconstruction, all previous works on MPI system matrix recovery are based on its sparsity when applying a Fourier transform or a discrete cosine transform (DCT). In this thesis, a representation of the system matrix in terms of low-rank tensors is proposed and embedded into an algorithm for the CS-based recovery of system matrices. In numerical experiments it is shown that the combination of transform-domain sparsity and low tensor rank yields system matrices with a higher quality and reduced noise.
- Finally, this work investigates the design of optimized sampling patterns for a given class of signals. Due to their clear link to statistical estimation theory, methods based on the Fisher information matrix (FIM) are particularly attractive for this type of problems. However, the resulting algorithms suffer from high numerical complexity and long running times. In this work, a new method based on the closely related Bayesian FIM is presented. Its main benefits are that it allows for a significant reduction of computation time. Moreover, it contains an additional channel to include information about the signals to be recovered. Its application both for MRI and MPI is demonstrated in numerical experiments.

Parts of this work were published in peer-reviewed journals [J1, J2] and presented at scientific conferences [C1–C5]. Moreover, several journal articles were published in collaboration with colleges from the University Medical Center Hamburg-Eppendorf [J3–J7], while other collaborative results were presented at conferences [C6–C11].

1.4 Structure of this work

This dissertation is organized in 9 chapters, which are grouped into three parts.

Part I contains an introduction to the mathematical framework used in this work and to the basic principles of MRI and MPI. Following this chapter, Chapter 2 introduces a general framework, which can be use to describe and analyze imaging experiments. After discussing the mathematical representation of images, a statistical description of imaging experiments is introduced. Making use of this description, we turn to the reconstruction of the underlying images, which can be performed using statistical estimation techniques. Finally, the basic principles of CS will be discussed and linked to the previously introduced framework.

The imaging modalities MPI and MRI share many similarities both with regard to their mathematical description and with regards to the underlying physical principles. These common principles are discussed in chapter 3 with an emphasis on the magnetization as the origin of the measured imaging signal.

Chapter 4 describes the physical principle as well as the spatial encoding used in MRI. Based on this, the MRI measurement model is introduced and linked to the mathematical framework introduced in chapter 2. Finally, it is shown how the most commonly used MRI image reconstruction techniques arise as special cases of the statistical methods from chapter 2.

Chapter 5 turns to the imaging modality MPI and describes the physical principles as well as the commonly used spatial encoding techniques. Based on this, the MPI system matrix is introduced to describe the relationship between image and measured signal. Subsequently, its general structure is discussed along with its measurement-based determination. Finally, the system matrix is used together with the estimation methods from chapter 2 in order to derive the most commonly used image reconstruction techniques.

Part II discusses the application of CS-based algorithms for image reconstruction as well as system calibration in MPI. The development of such algorithms strongly relies on suitable software frameworks, which provide the required data processing and optimization tools and which are both performant and easily extendable. To address this need, chapter 6 introduces the MRI reconstruction framework *MRIReco.jl* together with the MPI reconstruction framework *MPIReco.jl*. Both frameworks are written in the programming language julia. Moreover, both share the same optimization framework, which is also developed as part of this work. Validation of *MRIReco.jl* is performed using an open source dataset and a comparison against a state-of-the-art MRI reconstruction framework is performed.

In chapter 7, we consider MRI reconstruction in the presence of significant B_0-field inhomogeneity. For fast non-Cartesian sampling trajectories, these inhomogeneities are a serious cause of artifacts. As a matter of fact, taking them into account during image reconstruction leads to a large numerical cost. To minimize the computational cost, this chapter proposes an efficient approximation of the B_0-aware imaging operator. We combine it with a diagonal preconditioner, which operates directly in the MRI measurement space and effectively accelerates convergence of the used optimizer.

Finally, chapter 8 discusses the CS-based recovery of MPI system matrices. In particular, we introduce a new representation of the system matrix in terms of low-rank tensors. This characterization effectively complements the sparsity-based representations that are commonly used for system matrix recovery. To exploit it, we reformulate the system matrix recovery problem such that both the tensor rank and the number of sparse coefficients are minimized. The potential of this new method is demonstrated in numerical experiments.

Part III turns to the question of how to best choose sampling points either for the reconstruction of MRI images or the measurement of MPI system matrices. To this end, chapter 9 introduces a framework, which formulates the optimization of sampling points as an inverse problem based on the Bayesian Fisher information matrix. As a first application, chapter 9 shows how the derived method can be used to optimize MRI sampling patterns. Following this, chapter 10 shows how the proposed framework can be applied to find optimized sampling schemes for the sparse sampling of MPI system matrices.

Finally, chapter 11 summarizes the results of this work. Additionally, we discuss relevant questions remaining unanswered in this thesis and provide an outlook with regard to interesting projects that could further advance this interesting field of research.

Part I

Principles of MRI and MPI

2 Mathematical Background

This chapter serves to introduce the mathematical notation and the concepts used in this work for the description of imaging experiments and the subsequent image reconstruction.

2.1 Motivation

To motivate the selection of topics discussed in this chapter, Figure 2.1 illustrates a generic imaging experiment and highlights the central aspects that require mathematical modeling in order to obtain a useful description.

Images form the object of interest in this work because they provide a visual representation of the examined object in terms of the physical properties that the imaging device is sensitive to. Thus, the first problem to be addressed is that of image representation. Depending on the task at hand, either a continuous or a discrete image model turns out to be more suitable. Thus, after introducing the required mathematical preliminaries, we move on to a discussion of both continuous and discrete image representations.

Being able to describe images, the next task at hand is to understand the image formation process of the imaging device used. In abstract terms, imaging devices can be modeled as mappings that take an image as input and use it to compute the corresponding measurement data. Naturally, the measured data should be generated in a way such that inference of the underlying image is possible. In this work,

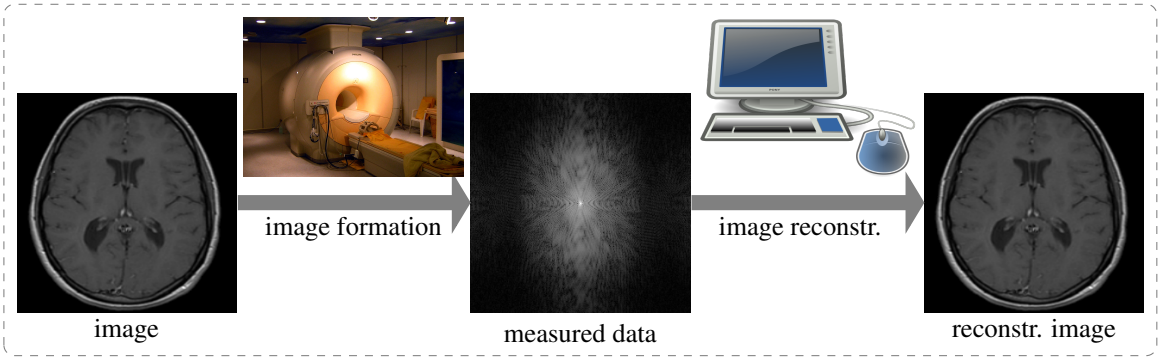


Figure 2.1: Illustration of a generic imaging experiment.

we mainly describe image formation using linear measurement mappings, as it turns out that these are sufficient to describe a large number of imaging modalities.

Having understood the image formation process, the last step in an imaging experiment is image reconstruction. Here one deals with the inverse problem of recovering the underlying image from the measured data. Image reconstruction thus involves finding some form of inverse for the image formation process. It can also be viewed as a statistical estimation problem. This latter approach is appealing in so far as it not only yields information about the object being imaged, but it can also provide uncertainty measures for the reconstructed image. For this reason our discussion of the image formation process is followed by a section on statistical estimation which introduces the relevant estimators commonly used for image reconstruction.

Having established a general framework for the description of imaging experiments, we next discuss the design of imaging experiments. A key requirement for the measured data is that it should contain enough information such that accurate inference of the underlying image is feasible. Ideally this should be achieved with a small number of measurements in order to minimize measurement time or radiation dose. To fulfill these requirements, experiment design was traditionally guided by the Nyquist-Shannon sampling theorem [18]. More recently, the paradigm of compressed sensing has demonstrated that image reconstruction can actually be performed from a significantly smaller number of measurements than required by the NSST given that the images are sparse in a known domain [19, 20].

Finally, we conclude this chapter with a discussion of error measures that can be used for the comparison of multiple images. These error measures play an important role in the development of new image reconstruction schemes because they allow us to compare the performance of different algorithms given the availability of datasets with known ground-truth images.

2.2 Mathematical Notation & Preliminaries

2.2.1 Basic Notation

In this work, standard mathematical notation is used for sets, vectors and matrices. In addition, we use the notation $\mathbb{K}$ to denote a general field, such as $\mathbb{R}$ or $\mathbb{C}$. We use small, boldface letters to denote vectors and capital, boldface letters to denote matrices. For instance, an N-dimensional, complex vector is written as

$$\mathbf{x} := (x_n)_{n=1,\dots,N} \in \mathbb{C}^N.$$

Similarly, a matrix of size $M \times N$ is denoted as

$$\mathbf{X} := (x_{m,n})_{m=1,\dots,M;n=1,\dots,N} \in \mathbb{C}^{M\times N}.$$

For convenience we also use indexed, boldface and lowercase letters $\mathbf{x}_{:,k}$ to denote the vector contained in the k^{th} column of the corresponding matrix $\mathbf{X}$. Similarly we let $\mathbf{x}_{k,:}$ denote the vector containing the

elements in the k^{th} row of $\mathbf{X}$. Following standard notation, the transposed of a $M \times N$ matrix $\mathbf{X}$ is written as

$$\mathbf{X}^{\mathsf{T}} := (x_{m,n})_{n=1,\ldots,N;m=1,\ldots,M} \in \mathbb{C}^{N\times M}.$$

and the adjoint is written as $\mathbf{X}^{\mathsf{H}} := (\mathbf{X}^{*})^{\mathsf{T}}$, with * denoting complex conjugation. Finally, the identity matrix is denoted as $\mathbf{1}$.

When dealing with random variables, we denote them using calligraphic letters. For instance, $\mathcal{X}$ denotes a scalar random variable and $\boldsymbol{\mathcal{X}}$ denotes a multivariate random variable. Related to this issue, we will use a hat to denote an estimator for a random variable. As an illustration, consider the vector-valued random variable $\boldsymbol{\mathcal{X}}$. In this work, a realization of this variable would be denoted by $\mathbf{x}$ and an estimator for it would be denoted by $\hat{\mathbf{x}}$.

2.2.2 Further Preliminaries

Before moving on to the mathematical description of imaging experiments, we use this section to collect some mathematical preliminaries that will be used throughout this work.

In the field of linear algebra, there exist multiple notions of a product, some of which will be used in this work. An important concept is the inner product between two vectors. For two vectors $\mathbf{x} \in \mathbb{C}^N$ and $\mathbf{y} \in \mathbb{C}^N$, it can be defined as

$$< \mathbf{x}, \mathbf{y} >= \mathbf{x}^{\mathsf{H}}\mathbf{y}. \tag{2.1}$$

This product can naturally be extended to matrices, which yields the well-known matrix product. Another form of product is the Kronecker product, which is a generalization of the so-called outer product. For two matrices $\mathbf{X} \in \mathbb{C}^{M\times N}$ and $\mathbf{Y} \in \mathbb{C}^{P\times Q}$, it is defined as

$$\mathbf{X} \otimes \mathbf{Y} = \begin{pmatrix} x_{1,1}\mathbf{Y} & \ldots & x_{1,N}\mathbf{Y} \\ \vdots & \ddots & \vdots \\ x_{M,1}\mathbf{Y} & \ldots & x_{M,N}\mathbf{Y} \end{pmatrix}. \tag{2.2}$$

Another important concept, which often reoccurs in this work, is the SVD. For a matrix $\mathbf{X} \in \mathbb{C}^{M\times N}$, the SVD is a decomposition of the form [45]

$$\mathbf{C} = \mathbf{U}\mathrm{diag}\,(\boldsymbol{\sigma})\,\mathbf{V}^{\mathsf{H}}. \tag{2.3}$$

Here $\mathbf{U} \in \mathbb{C}^{M\times M}$ and $\mathbf{V} \in \mathbb{C}^{N\times N}$ are unitary matrices containing the left- and right singular vectors in their respective columns. Moreover, $\mathrm{diag}\,(\boldsymbol{\sigma}) \in \mathbb{C}^{M\times N}$ is a rectangular matrix containing the singular values $\boldsymbol{\sigma} \in \mathbb{R}_{+}^{\min(M,N)}$ in non-increasing order. An important property of the SVD is that the number of non-zero singular values corresponds exactly to the matrix rank, i.e. the dimensionality of the space

spanned by the matrix rows or columns. Beyond that, the SVD is often used to compute low-rank approximations. This is based on the property that the truncated matrix SVD

$$\mathbf{X}_K = \mathbf{U}\mathrm{diag}\left(\sigma_1, \dots, \sigma_K, 0, \dots\right)\mathbf{V}^{\mathsf{H}} \tag{2.4}$$

provides the optimal rank-K approximation to the underlying matrix $\mathbf{X}$ in the sense that $\mathbf{X}_K$ is the rank-K matrix with the minimal ℓ_2-distance to $\mathbf{X}$ [45].

Finally, we use this section to introduce the Fourier transform, which allows to decompose a signal into its constituent frequency components. In this work, we use a tilde to denote Fourier transformed quantities. Thus, the discrete Fourier transform (DFT) of a time signal $\mathbf{x} \in \mathbb{C}^N$ will be denoted by the vector $\tilde{\mathbf{x}}$ whose k^{th} component is given by

$$\tilde{x}_k = \frac{1}{\sqrt{N}} \sum_{n=1}^{N} x_n \mathrm{e}^{-2\pi \mathrm{i} f_k t_n}. \tag{2.5}$$

Similarly the Fourier transform of a continuous time signal $x : \mathbb{R} \to \mathbb{R}$ will be denoted by

$$\tilde{x}(f) = \int_{\mathbb{R}} x(t) \mathrm{e}^{-2\pi \mathrm{i} f t} \, \mathrm{d}t. \tag{2.6}$$

2.3 Representation of Images

The imaging modalities considered in this work aim to visualize the distribution of signal generating particles. For instance, MRI provides information on the distribution of nuclear spins in the imaged sample and MPI maps the concentration of SPIOs. Following the concepts of classical physics, such distributions can be described quite generically using an image function

$$f : V \subset \mathbb{R}^3 \to \mathbb{K}, \quad \mathbf{r} \mapsto f(\mathbf{r}). \tag{2.7}$$

Depending on the context, an image can be either real-valued ($\mathbb{K} = \mathbb{R}$) or complex-valued ($\mathbb{K} = \mathbb{C}$). A second characteristic of images is that they are typically determined within a compact volume of interest $V \subset \mathbb{R}^3$, which is also termed the field of view (FOV). Additionally, we assume the images to be sufficiently well behaved in the sense that they are integrable and square integrable, i.e. $f \in L^1(V) \cap L^2(V)$. Visualization of images is usually done by displaying 2d slices of the acquired image. These 2d images can be viewed as a special case by considering 2d hyperplanes in $\mathbb{R}^3$.

For visualization and processing of images on the computer, one usually works with a discrete approximation of the continuous-space image representation described above. To obtain such a representation, we assume V to be a cube centered at the origin of the coordinate system

$$V = [-l_x/2, l_x/2] \times [-l_y/2, l_y/2] \times -l_z/2, l_z/2] \tag{2.8}$$

and discretize it using a uniform grid of size $N = N_x \times N_y \times N_z$. The associated grid points can be written as

$$\mathbf{r}_{n_x,n_y,n_z} = \begin{pmatrix} -l_x/2 + (n_x - 1)\Delta_x \\ -l_y/2 + (n_y - 1)\Delta_y \\ -l_z/2 + (n_z - 1)\Delta_z \end{pmatrix}, \tag{2.9}$$

with

$$\Delta_x = \frac{l_x}{N_x - 1},\ \Delta_y = \frac{l_y}{N_y - 1},\ \Delta_z = \frac{z_z}{N_z - 1},$$

a voxel size of $\Delta V = \Delta_x \Delta_y \Delta_z$ and $n_{x,y,z} \in \{1, \dots, N_{x,y,z}\}$. Using this grid, a continuous image $f(\mathbf{r})$ can be approximated by a 3-way tensor $\mathbf{f} \in \mathbb{K}^{N_x \times N_y \times N_z}$ with elements

$$f_{n_x,n_y,n_z} = f(\mathbf{r}_{n_x,n_y,n_z}). \tag{2.10}$$

For the mathematical notation it is often convenient to traverse the 3d grid using a linear index. To achieve this, we associate each 3-tuple (n_x, n_y, n_z) with a linear index $n = n_x + (N_x - 1)n_y + (N_x - 1)(N_y - 1)n_z$. Hence, we obtain an equivalent representation of f as the vector

$$\mathbf{f} = (f(\mathbf{r}_n))_{n=1,\dots,N}.$$

This representation gives us a unified way of describing images, irrespective of their dimensionality. Importantly, linear transformations of the image can be written as simple matrix-vector products

$$\mathbf{y} = \mathbf{H}\mathbf{f}, \tag{2.11}$$

where $\mathbf{H} \in \mathbb{K}^{M \times N}$ describes the linear transformation at hand. This way of writing linear mappings will prove quite useful in the remainder of this work. For example, we shall see that the signal encoding in both MRI and MPI can be described using linear transformations.

2.4 Imaging Experiments

A common aspect shared by many imaging modalities is that the image $f(\mathbf{r})$ and the discrete measurement signal $\mathbf{y} \in \mathbb{K}^M$ are related by a linear relationship. To describe such measurements, we let $\{h_j : \mathbb{R}^3 \to \mathbb{K} \,|\, j = 1, \dots, N\}$ denote a set of available measurement basis functions. For future reference, we also introduce the set $\Gamma = \{1, \dots, N\}$, which contains the indices of all possible measurements and the set $\Omega \subseteq \Gamma$ containing the indices of all measurements actually performed. Using this notation, one obtains a discrete set of measurements which can be modeled as

$$y_j = \int_V h_j(\mathbf{r}) f(\mathbf{r}) \, \mathrm{d}^3 r + \eta_j. \tag{2.12}$$

Here, the ideal measurement, described by the basis function h_j, is superimposed by a stochastic component $\eta_j \in \mathbb{K}$. The latter describes measurement noise of varying origins, which corrupts the ideal measurements and which is present in most practical applications. The linear measurement model can also be discretized in the image domain by using the grid points (2.9) and applying a rectangular rule to approximate the integral. If we ignore the resulting approximation error and if we collect all measurements in the vector $\mathbf{y}$ and the noise in the vector $\boldsymbol{\eta}_\Omega$, the resulting discrete model takes the form

$$\mathbf{y} = \mathbf{H}_\Omega \mathbf{f} + \boldsymbol{\eta}_\Omega, \tag{2.13}$$

with the measurement matrix $\mathbf{H}_\Omega = (h_j(\mathbf{r}_i)\Delta V)_{j\in\Omega;i=1,\dots,N}$.

A very common measurement basis is the canonical basis $h_j(\mathbf{r}) = \delta(\mathbf{r} - \mathbf{r}_j)$, where each sample corresponds to a measurement at one spatial location. As an example, such a measurement basis can be used to model optical cameras, where a sensor captures the signal associated with each image pixel. Moreover, it plays an important role in MPI, where it can be used to model the robot-based measurement of the MPI system matrix. This will be elaborated further in chapter 5. Upon discretization, the corresponding measurement matrix takes the form of a sampling matrix

$$\mathbf{H}_\Omega = (\delta_{j,i})_{j\in\Omega;i=1,\dots,N}\,, \tag{2.14}$$

where $\delta_{j,i}$ denotes the Kronecker delta. Thus, each row contains a one at the index of the voxel being sampled and zeros elsewhere.

Another popular measurement basis is the Fourier basis $h_j(\mathbf{r}) = {}^1\!/\!\sqrt{N}\mathrm{e}^{-2\pi\mathrm{i}\mathbf{k}_j\cdot\mathbf{r}}$, where each measurement samples one coefficient of the Fourier transform of the underlying image. This form of Fourier encoding commonly arises when a spectral signal is sampled in the time domain but it also plays an important role in MRI, as discussed in chapter 4. Upon discretization, the corresponding measurement matrix takes the form of a discrete Fourier transform

$$\mathbf{H}_\Omega = \left(\frac{1}{\sqrt{N}}\mathrm{e}^{-2\pi\mathrm{i}\mathbf{k}_j\cdot\mathbf{r}_i}\right)_{j\in\Omega;i=1,\dots,N}. \tag{2.15}$$

The distribution of the measurement noise depends strongly on the application at hand. For instance, the noise in x-ray and CT imaging is known to be well described by a Poisson distribution. In many other applications, normally distributed measurement noise $\boldsymbol{\eta} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_\mathbf{y})$ with mean $\mathbf{0}$ and noise-covariance matrix $\boldsymbol{\Sigma}_\mathbf{y} \in \mathbb{K}^{M\times M}$ is a good model. For instance, such a description is known to hold for thermal noise, which corrupts the voltage induced in MRI receive coils. For a broad range of applications, the use of this model can also be motivated by the central limit theorem. Loosely speaking, the latter asserts that in many situations, the distribution of the sum of multiple independent variables tends towards a normal distribution [46]. Thus, normally distributed noise is often a good model, when the measurement noise is an additive combination of a sufficiently large number of independent noise sources. Finally, the

assumption of normally distributed noise is very popular due to the convenient form of the associated statistical estimators.

2.5 Statistical Estimation

The previous sections have introduced our framework for the description of images and of the measurements performed in imaging experiments. We now turn to the inverse problem of recovering the image at hand from the measurements performed in the experiment. As imaging data is typically corrupted by measurement noise, it makes sense to approach image reconstruction from the point of view of statistics. Using statistical estimation techniques does not only enable us to obtain estimates for the image underlying an imaging experiments, but it also allows us to characterize the expected uncertainty of the chosen image reconstruction method. This opens the door to further optimization of the measurement scheme used for imaging.

2.5.1 Maximum Likelihood Estimation

Having performed a measurement as described in section 2.4, image reconstruction is concerned with the estimation of the underlying image $\mathbf{f}$ given the noisy measurements $\mathbf{y}$. Typically, this is done by computing a suitable statistical estimator. Here a statistical estimator is a function that maps from the set of possible measurements to the set of estimates. For example, for the noisy linear measurement model (2.13), an estimator is a function

$$\hat{\mathbf{f}} : \mathbb{K}^M \to \mathbb{K}^N, \quad \mathbf{y} \mapsto \hat{\mathbf{f}}(\mathbf{y}).$$

A commonly used estimator is the maximum likelihood (ML) estimator. The latter searches for the parameter $\mathbf{f}$ that maximizes the likelihood $p(\boldsymbol{\mathcal{Y}} = \mathbf{y} | \, \boldsymbol{\mathcal{F}} = \mathbf{f})$ of the observed measurements $\mathbf{y}$. Thus, the ML estimator can be written as [47]

$$\hat{\mathbf{f}}_{\mathrm{ML}}(\mathbf{y}) = \operatorname*{argmax}_{\mathbf{f}} p(\boldsymbol{\mathcal{Y}} = \mathbf{y} | \, \boldsymbol{\mathcal{F}} = \mathbf{f}). \tag{2.16}$$

For illustration, let us consider the case of normally distributed measurement noise, which was introduced in the last section. If we assume prior knowledge of the noise covariance matrix $\boldsymbol{\Sigma}_{\mathbf{y}}$ the ML estimator can be written as

$$\hat{\mathbf{f}}_{\mathrm{ML}}(\mathbf{y}) = \operatorname*{argmax}_{\mathbf{f}} \frac{1}{\sqrt{(2\pi)^N \det(\boldsymbol{\Sigma}_{\mathbf{y}})}} \exp\Big(-\frac{1}{2} (\mathbf{y} - \mathbf{H}_\Omega \mathbf{f})^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathbf{y}}^{-1} (\mathbf{y} - \mathbf{H}_\Omega \mathbf{f}) \Big).$$

To simplify this problem, we note that one can equivalently minimize the negative logarithm of the likelihood. By doing so one ends up with

$$\hat{\mathbf{f}}_{\mathrm{ML}}(\mathbf{y}) = \underset{\mathbf{f}}{\operatorname{argmin}}\ (\mathbf{y} - \mathbf{H}_\Omega \mathbf{f})^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathbf{y}}^{-1} (\mathbf{y} - \mathbf{H}_\Omega \mathbf{f}). \tag{2.17}$$

The first order optimality condition for this problem is

$$\mathbf{H}_\Omega^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathbf{y}}^{-1} \mathbf{H}_\Omega \mathbf{f} = \mathbf{H}_\Omega^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathbf{y}}^{-1} \mathbf{y},$$

which immediately leads to the following form of the ML estimator

$$\hat{\mathbf{f}}_{\mathrm{ML}}(\mathbf{y}) = (\mathbf{H}_\Omega^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathbf{y}}^{-1} \mathbf{H}_\Omega)^{-1} \mathbf{H}_\Omega^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathbf{y}}^{-1} \mathbf{y}. \tag{2.18}$$

A commonly encountered special case is that of a diagonal noise covariance matrix of the form $\boldsymbol{\Sigma}_{\mathbf{y}} = \sigma^2 \mathbf{1}$. In this case, the ML estimator is the solution to the ordinary least-squares problem

$$\hat{\mathbf{f}}_{\mathrm{ML}}(\mathbf{y}) = \underset{\mathbf{f}}{\operatorname{argmin}}\ \|\mathbf{y} - \mathbf{H}_\Omega \mathbf{f}\|_2^2 \tag{2.19}$$

and it can be written as

$$\hat{\mathbf{f}}_{\mathrm{ML}}(\mathbf{y}) = (\mathbf{H}_\Omega^{\mathsf{H}} \mathbf{H}_\Omega)^{-1} \mathbf{H}_\Omega^{\mathsf{H}} \mathbf{y} = \mathbf{H}_\Omega^{\dagger} \mathbf{y}. \tag{2.20}$$

Here the matrix $\mathbf{H}_\Omega^{\dagger} = (\mathbf{H}_\Omega^{\mathsf{H}} \mathbf{H}_\Omega)^{-1} \mathbf{H}_\Omega^{\mathsf{H}}$ is the Moore-Penrose pseudoinverse of the measurement matrix $\mathbf{H}_\Omega$ [48].

When performing maximum likelihood estimation, the expected accuracy of the ML estimator strongly depends on the signal encoding operator $\mathbf{H}_\Omega$. In order to quantify the expected estimation error, one can compute the covariance matrix of the estimator used. For the ML estimator applied to the linear measurement model with Gaussian noise, one can easily show that the corresponding covariance matrix is given by [47]

$$\operatorname{cov}(\hat{\mathbf{f}}_{\mathrm{ML}}) = \left(\mathbf{H}_\Omega^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathbf{y}}^{-1} \mathbf{H}_\Omega\right)^{-1}. \tag{2.21}$$

The ML estimator is a popular choice for image reconstruction for multiple reasons. For the case of Gaussian measurement noise, the ML estimator is unbiased. Moreover, it is optimal in the sense that its corresponding covariance matrix coincides with the Cramer-Rao bound (CRB) of the associated statistical model [47]. This last result establishes that for any other unbiased estimator $\hat{\mathbf{f}}$, the associated covariance matrix can be lower bounded in the following way

$$\operatorname{cov}(\hat{\mathbf{f}}) \succeq \left(\mathbf{H}_\Omega^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathbf{y}}^{-1} \mathbf{H}_\Omega\right)^{-1} = \operatorname{cov}(\hat{\mathbf{f}}_{\mathrm{ML}}). \tag{2.22}$$

Here the notation $\succeq$ refers to the Loewner semi-ordering. Hence, the meaning of (2.22) is that $\operatorname{cov}(\hat{\mathbf{f}}) - \operatorname{cov}(\hat{\mathbf{f}}_{\mathrm{ML}})$ is a positive semi-definite matrix, i.e. $\mathbf{v}^H \left(\operatorname{cov}(\hat{\mathbf{f}}) - \operatorname{cov}(\hat{\mathbf{f}}_{\mathrm{ML}})\right) \mathbf{v} \geq 0$ for any $\mathbf{v} \in \mathbb{K}^N$.

Importantly, this implies that the diagonal elements of $\mathrm{cov}(\hat{\mathbf{f}})$ can be lower bounded by the corresponding elements for the ML estimator, i.e. $\mathrm{cov}(\hat{\mathbf{f}})_{ii} \geq \mathrm{cov}(\hat{\mathbf{f}}_{\mathrm{ML}})_{ii}$.

A drawback of the ML estimator is its instability when the image reconstruction problem is ill-conditioned. For instance, the ML estimator for the Gaussian case (2.19) is known to exhibit noise amplification when $\mathbf{H}_\Omega$ has a large condition number.

2.5.2 Maximum A-Posteriori Estimation

As discussed in the last section, ML estimation can become unstable when the corresponding inverse problem is ill-conditioned. A popular way to alleviate such issues is to incorporate prior knowledge about the structure of the underlying signal that is to be recovered. In Bayesian statistics this form of prior knowledge is incorporated by specifying a prior distribution, which reflects ones prior expectation about the parameters to be estimated. While being a powerful tool, this also leads to a different interpretation of the underlying signal of interest. ML estimation is based on a frequentist interpretation, where one assumes the existence of one true parameter $\mathbf{f}$ whose measurement generates the observed data $\mathbf{y}$. In contrast, the Bayesian approach assumes $\mathbf{f}$ to be drawn from the prior distribution $p(\boldsymbol{\mathcal{F}} = \mathbf{f})$. The subsequently performed measurements then allow us to perform inference on $\mathbf{f}$ based on both the measurements and our prior belief. For the analysis of the measured data $\mathbf{y}$, this implies that expectation values are computed with respect to the joint distribution of $\mathbf{f}$ and $\mathbf{y}$.

For illustration, let us consider the image reconstruction problem discussed in the last section. Without having performed a measurement, ones expectations about the underlying image are solely determined by the prior distribution $p(\boldsymbol{\mathcal{F}} = \mathbf{f})$. Having performed a set measurements $\mathbf{y}$, the expectations about $\mathbf{f}$ can be updated by using Bayes rule

$$p(\boldsymbol{\mathcal{F}} = \mathbf{f}|\boldsymbol{\mathcal{Y}} = \mathbf{y}) = \frac{p(\boldsymbol{\mathcal{Y}} = \mathbf{y}|\boldsymbol{\mathcal{F}} = \mathbf{f})p(\boldsymbol{\mathcal{F}} = \mathbf{f})}{\int_{\mathbb{K}^N} p(\boldsymbol{\mathcal{Y}} = \mathbf{y}|\boldsymbol{\mathcal{F}} = \mathbf{f}')p(\boldsymbol{\mathcal{F}} = \mathbf{f}')d^N f'}. \tag{2.23}$$

Here the numerator contains the prior $p(\boldsymbol{\mathcal{F}} = \mathbf{f})$ and the likelihood of the data for the given estimate $p(\boldsymbol{\mathcal{Y}} = \mathbf{y}|\boldsymbol{\mathcal{F}} = \mathbf{f})$. The distribution in the denominator can be interpreted as a normalization factor and is also called the marginal distribution of $\boldsymbol{\mathcal{Y}}$. The resulting distribution is called the posterior distribution for $\boldsymbol{\mathcal{F}}$ and it describes how ones expectations about $\boldsymbol{\mathcal{F}}$ should be updated given the information contained in the performed measurements. Importantly, one can obtain an estimate for the underlying image, by choosing $\mathbf{f}$ such that the corresponding value of the posterior distribution is maximized. The resulting estimator is called the maximum a posteriori (MAP) estimator.

When performing image reconstruction for the noisy, linear model (2.13), the most commonly chosen prior is a Gaussian distribution $\mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_\mathbf{f})$ with covariance matrix $\boldsymbol{\Sigma}_\mathbf{f}$. This expresses the expectation that the parameter values in $\mathbf{f}$ should be small with the scale being determined by $\boldsymbol{\Sigma}_\mathbf{f}$. Computing the

posterior for this prior distribution, one finds that the MAP estimate is the solution of the optimization problem

$$\hat{\mathbf{f}}_{\text{MAP}}(\mathbf{y}) = \underset{\mathbf{f}}{\operatorname{argmin}} \, (\mathbf{y} - \mathbf{H}_\Omega \mathbf{f})^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathbf{y}}^{-1} (\mathbf{y} - \mathbf{H}_\Omega \mathbf{f}) + \mathbf{f}^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathbf{f}}^{-1} \mathbf{f}. \tag{2.24}$$

In analogy to the ML estimator, the first order optimality condition for this MAP estimator is

$$(\mathbf{H}_\Omega^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathbf{y}}^{-1} \mathbf{H}_\Omega + \boldsymbol{\Sigma}_{\mathbf{f}}^{-1}) \mathbf{f} = \mathbf{H}_\Omega^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathbf{y}}^{-1} \mathbf{y}.$$

Based on this, the MAP estimator can be written as

$$\hat{\mathbf{f}}_{\text{MAP}}(\mathbf{y}) = (\mathbf{H}_\Omega^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathbf{y}}^{-1} \mathbf{H}_\Omega + \boldsymbol{\Sigma}_{\mathbf{f}}^{-1})^{-1} \mathbf{H}_\Omega^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathbf{y}}^{-1} \mathbf{y}. \tag{2.25}$$

Next, let us again consider the special case of a diagonal noise covariance matrix $\boldsymbol{\Sigma}_{\mathbf{y}} = \sigma_{\mathbf{y}} \mathbf{1}$. If we additionally assume a diagonal prior covariance matrix of the form $\boldsymbol{\Sigma}_{\mathbf{f}} = \sigma_{\mathbf{f}}^2 \mathbf{1}$, the MAP estimator is the solution of the classical Tikhonov-regularized least squares problem

$$\hat{\mathbf{f}}_{\text{MAP}}(\mathbf{y}) = \underset{\mathbf{f}}{\operatorname{argmin}} \, \|\mathbf{y} - \mathbf{H}_\Omega \mathbf{f}\|_2^2 + \lambda \|\mathbf{f}\|_2^2, \tag{2.26}$$

with $\lambda = \frac{\sigma_{\mathbf{y}}^{2M}}{\sigma_{\mathbf{f}}^{2N}}$. By comparison with (2.19), we see that the resulting MAP estimator correspond to the ML estimator except for the additional penalty term $\lambda \|\mathbf{f}\|_2^2$, which prevents the solution from becoming too large. The penalty term also implies that the MAP estimator is biased, which is in contrast to the respective ML estimator. Nevertheless, it often turns out beneficial to tolerate a certain amount of bias in order to mitigate the larger variances which often occur in ML estimation.

To obtain an estimate for the uncertainty associated with the MAP estimator, one can again compute the corresponding covariance matrix. In Bayesian statistics this is usually done under the assumption that the parameters $\mathbf{f}$ were indeed sampled from the assumed prior distribution. Doing so for the Gaussian prior $\mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\mathbf{f}})$, one can show that the error covariance matrix has the form [47]

$$\operatorname{cov}(\hat{\mathbf{f}}_{\text{MAP}}) = \left(\boldsymbol{\Sigma}_{\mathbf{f}}^{-1} + \mathbf{H}_\Omega^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathbf{y}}^{-1} \mathbf{H}_\Omega \right)^{-1}. \tag{2.27}$$

Note that this expression corresponds exactly to the covariance matrix for the ML estimator, except for the additional term containing the prior covariance matrix.

While the Gaussian prior is the most popularly used prior for the linear Gaussian measurement model, other priors are also in common use. As an example, consider prior distributions of the form $p(\mathbf{f}) \propto \exp(-\rho \|\boldsymbol{\Psi} \mathbf{f}\|_\ell^\ell)$, where $\boldsymbol{\Psi}$ is a linear transformation and $\rho > 0$. If we again assume Gaussian measurement noise with covariance matrix $\boldsymbol{\Sigma}_{\mathbf{y}} = \sigma_{\mathbf{y}} \mathbf{1}$, it is easy to see that the corresponding MAP estimator is equal to

$$\hat{\mathbf{f}}_{\text{MAP}}(\mathbf{y}) = \underset{\mathbf{f}}{\operatorname{argmin}} \, \|\mathbf{y} - \mathbf{H}_\Omega \mathbf{f}\|_2^2 + \lambda \|\boldsymbol{\Psi} \mathbf{f}\|_\ell^\ell, \tag{2.28}$$

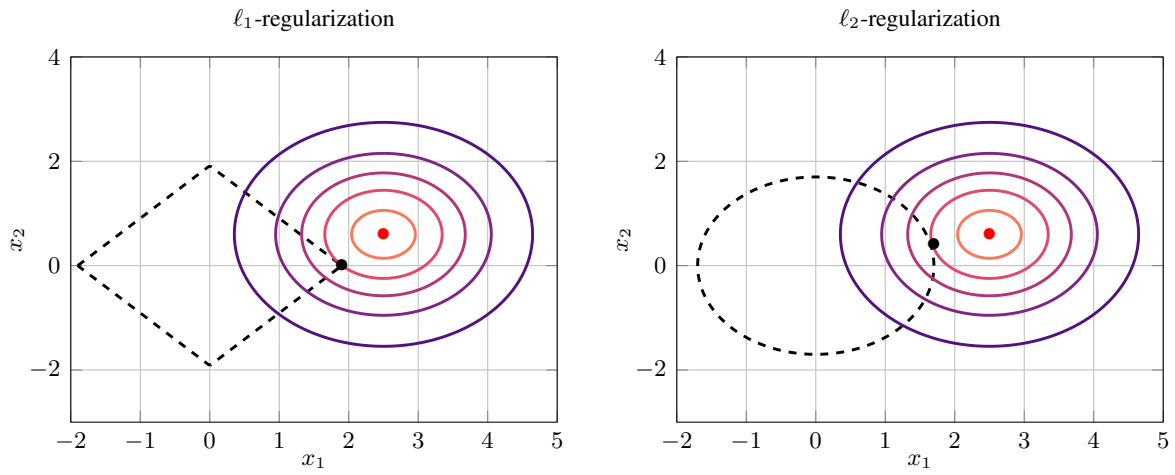


Figure 2.2: Illustration of ℓ_1-regularization (left) and ℓ_2-regularization (right). The red dot marks a measurement subject to Gaussian noise. The corresponding likelihood is indicated by the colored contour plot and the dashed line shows an isosurface of the respective norm to be minimized. The black dots illustrate estimates both having the same likelihood, but each minimizing the respective norm.

with $\lambda = \frac{\rho}{2\pi\sigma_{\mathbf{y}}^{2M}}$. In this work, the case $\ell = 1$ is of special interest. Here the prior distribution corresponds to a Laplace distribution and the MAP estimator reduces to the solution of an ℓ_1-regularized least squares problem. The resulting ℓ_1-regularization term plays an important role in the field of compressed sensing because it encourages solutions for which $\mathbf{\Psi f}$ is sparse. This is illustrated in Figure 2.2, where we consider a 2d measurement corrupted by Gaussian measurement noise. As shown in the right panel, minimizing the ℓ_2-norm subject to a constrained on the likelihood yields a solution for which both components are shrank but non-zero. On the other hand, minimizing the ℓ_1-norm favors sparse solutions as is shown in the left panel.

2.6 Nyquist Sampling

When designing an imaging experiment, one naturally arrives at the question, how many and which kinds of measurements should be performed in order to be able to guarantee a successful reconstruction of the underlying image. The most commonly used theorem to answer these questions is the Nyquist-Shannon sampling theorem, which relates the spacing of sampling points to the maximum frequency contained in the sampled signal [18].

Theorem 2.1. *Let $f \in L^2(\mathbb{R})$ be a square-integrable function whose Fourier transform $\tilde{f}$ is zero for all frequencies ν fulfilling $|\nu| > \nu_{max}$. Then f can be uniquely reconstructed from a set of equidistant sampling points whose spacing does not exceed $\frac{1}{2\nu_{max}}$.*

The theorem was formulated in a 1d setting, but it can be naturally extended into higher dimensions by analyzing the corresponding multidimensional frequency vectors. If an estimate for the maximum frequency in a signal is available, a suitable measurement scheme can easily be found based on Theorem 2.1.

A strength of the NSST is its wide applicability, which results from the weak assumptions made by it. A drawback is that it only gives an upper bound for the number of measurements required to guarantee reconstructability of a signal. Thus, it seems likely that successful reconstruction of some signals should be possible from a smaller number of samples, given that some form of prior knowledge about the signal is provided.

2.6.1 Image Space Sampling

To illustrate the application of the NSST, let us first consider a bandlimited signal $f_{1\mathrm{d}} : \mathbb{R} \to \mathbb{K}$ with a maximum frequency $\nu_{\max}$. In line with our discussion on images, we would also like $f_{1\mathrm{d}}$ to be supported only on an interval $[-l_x/2, l_x/2]$. As a matter of fact, the assumptions on bandlimitedness and compact support, cannot be simultaneously fulfilled in reality, due to the properties of the Fourier transform. Therefore, we make the weaker assumption, that the values of $f_{1\mathrm{d}}$ outside the interval $[-l_x/2, l_x/2]$ are sufficiently small, such that they can be neglected for practical purposes.

If the signal is to be sampled using the canonical measurement basis, Theorem 2.1 ascertains that the underlying signal can be reconstructed given that the sampling points have a spacing $\Delta_x \leq \frac{1}{2\nu_{\max}}$. Based on this, a minimum set of sampling points consistent with the NSST can be written as

$$\left\{ x_{n_x} = -l_x/2 + \frac{n_x - 1}{2\nu_{\max}} \;\middle|\; n_x = 1, \ldots, N_x \right\},$$

with $N_x = 2\nu_{\max} l_x + 1$.

Based on our discussion on 1d-signals, the sampling of 2d and 3d images can be treated by applying the results from the 1d-case to each dimension of the image at hand. As an example, consider an image $f : V \to \mathbb{K}$ localized within the cube V specified in (2.8) and let $\nu_{\max}$ be the maximum spatial frequency in all spatial dimensions. In this case a suitable set of sampling points are the grid points (2.9) with $\Delta_{x,y,z} = \frac{1}{2\nu_{\max}}$. The number of required grid points follows directly from the requirement that the sampling points should cover the whole FOV. Thus, $N_{x,y,z} = 2\nu_{\max} l_{x,y,z} + 1$. Sampling the image with a smaller frequency leads to artifacts such as the common partial-volume artifact.

2.6.2 Fourier Space Sampling

Similar to the previous case, the NSST can also be used for measurements performed in the Fourier measurement basis. To derive the corresponding results, let us again consider the 1d-signal $f_{1\mathrm{d}}$ from the previous section. In this case, each measurement performed provides us with one sample of the Fourier

transformed signal $\tilde{f}_{1\mathrm{d}}$. By means of the inverse Fourier transform, an exact reconstruction of $f_{1\mathrm{d}}$ can be obtained, given that the measured Fourier samples allow an exact reconstruction of $\tilde{f}_{1\mathrm{d}}$.

To apply the NSST to $\tilde{f}_{1\mathrm{d}}$, we first note that its Fourier transform is indeed supported on the interval $[-l_x/2, l_x/2]$. Thus, neighboring samples of $\tilde{f}_{1\mathrm{d}}$ should be separated no further than $\Delta k_x = 1/l_x$. Furthermore, the acquired Fourier samples should cover the full support of $\tilde{f}_{1\mathrm{d}}$. By combining these requirements, one finds the following minimal set of Fourier coordinates, for which the NSST guarantees the reconstructability of $f_{1\mathrm{d}}$

$$\left\{ k_{n_x} = -\nu_{\max} + \frac{n_x - 1}{l_x} \;\middle|\; n_x = 1, \ldots, N_x \right\},$$

with $N_x = 2\nu_{\max} l_x + 1$. Note that the number of grid points N_x is exactly the same for the canonical measurement basis and the Fourier measurement basis.

Similar to the last section, the 1d example can be readily generalized in order to deal with 2d or 3d images. Thus, if we consider the general 3d signal f, a set of Fourier nodes fulfilling the assumptions of the NSST is given by

$$\mathbf{k}_{n_x,n_y,n_z} = \begin{pmatrix} -\nu_{\max} + \frac{n_x-1}{l_x} \\ -\nu_{\max} + \frac{n_y-1}{l_y} \\ -\nu_{\max} + \frac{n_z-1}{l_z} \end{pmatrix}, \tag{2.29}$$

with $n_{x,y,z} \in [1, N_{x,y,z}]$. Here N_y and N_z are defined completely analogously to N_x.

2.7 Compressed Sensing

Compressed Sensing is a new sampling paradigm, which allows the recovery of signals under conditions where the Nyquist-Shannon sampling theorem is not applicable. To be specific, CS is concerned with signals that have a sparse representation in a known transform domain. CS theory establishes that such signals can be recovered from significantly fewer samples than required by Theorem 2.1 given that a suitable measurement scheme is used. If this is the case, CS theory ascertains that the underlying signal can be recovered by searching for the sparsest signal in the sparsity domain that is consistent with the measured data.

A reduction of measurements as just described can be desirable for multiple reasons. For instance, reducing the number of measurements can lead to a decrease in measurement time for time-critical applications. In other situations, the sensors required for a measurement might be expensive. In this case, reducing the number of measurements can lead to a reduction of costs for the experiment.

2.7.1 Sparsity

The first central assumption of CS is that the signal $\mathbf{f}$ should be sparse in a known transform domain. To formalize this, let $\mathbf{\Psi} \in \mathbb{K}^{N \times N}$ be a suitable sparsifying transform and let $\mathbf{\Psi}^\dagger$ denote a pseudo-inverse

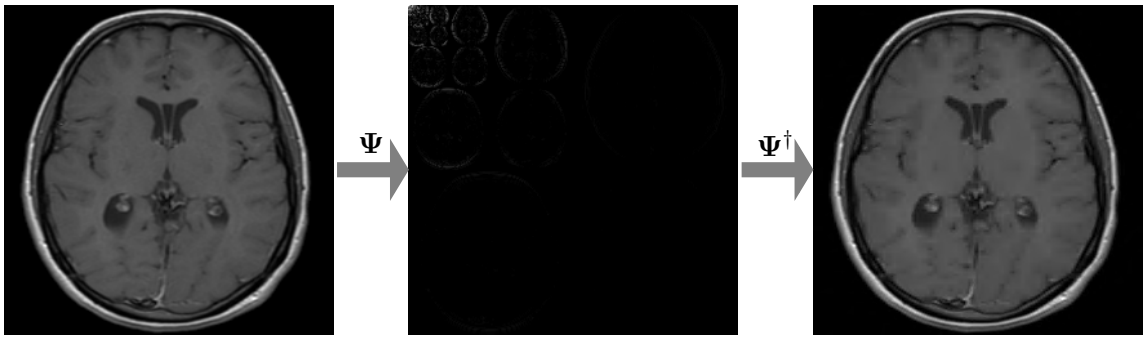


Figure 2.3: Illustration of image compressibility. The original image (left panel) is brought into the sparsity domain by application of a Wavelet transform. The sparse representation (middle panel in log-scale) is compressed by retaining only the 10% largest coefficients. The compressed image (right panel) is reconstructed by applying an inverse Wavelet transformation.

such that $\mathbf{\Psi}^\dagger \mathbf{\Psi} \mathbf{f} = \mathbf{f}$ for all $\mathbf{f} \in \mathbb{K}^N$. For CS to be applicable we assume the vector $\boldsymbol{\alpha} = \mathbf{\Psi} \mathbf{f}$ to contain at most $S \ll N$ non-zero entries, i.e. to be S-sparse. In this setting, the noise-free measured data can also be written as

$$\mathbf{y} = \mathbf{H}_\Omega \mathbf{\Psi}^\dagger \boldsymbol{\alpha}$$

with $\mathbf{H}_\Omega \mathbf{\Psi}^\dagger$ playing the role of a generalized measurement matrix. Focusing on the sparse vector $\boldsymbol{\alpha}$ thus greatly reduces the size of the solution space because a large number of the components of $\boldsymbol{\alpha}$ can be assumed to be zero. Hence, it appears quite natural that one might be able to reduce the number of samples if the reconstruction method forces the solution to be sparse in the corresponding transform domain.

We note that in practical applications, sparsity is often not strictly fulfilled. However, many real signals are compressible. This means that one can find an approximation $\mathbf{f}_S$ so that $\mathbf{\Psi} \mathbf{f}_S$ is S-sparse and the differences between $\mathbf{f}$ and $\mathbf{f}_S$ become negligibly small. For images, this compressibility is a well known property, which is commonly used by lossy image formats. For instance, JPEG-2000 applies a Wavelet transform to the image at hand before quantizing it. Efficient storage of images is achieved by only storing the largest coefficients. In this way, 5 to 10 fold compression can often be achieved without compromising the visual quality of images [49]. This is illustrated in Figure 2.3, where a Wavelet transformation was applied to a test image and compression was performed by only retaining the 10% largest coefficients.

An illustration of the central operators and domains used in CS is provided in Figure 2.4(a)-(d) for the case of a sparse 1d signal. The signal of interest (shown in panel (b)) is sampled using the Fourier measurement basis (2.15), which gives rise to the measurements shown in panel (a). The sparsified signal is shown in panel (c). Since the signal itself is sparse, the canonical measurement basis (2.14) can be used as the sparsifying transform $\mathbf{\Psi}$. CS uses both transformations along with the knowledge about the signal's sparsity to recover the underlying signal, which is illustrated in panel (d).

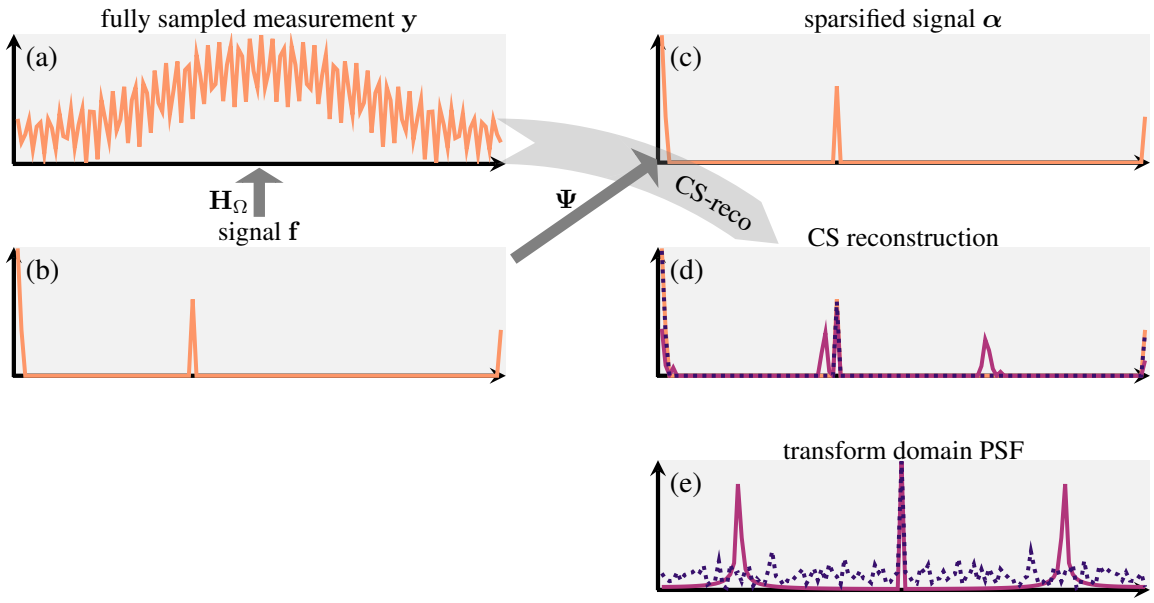


Figure 2.4: Illustration of the domains and operations relevant for CS. The sparse 1d-signal in panel (b) is sampled using the Fourier measurement basis as shown in panel (a). The sparsifying transform corresponds to the identity since the signal itself is sparse. Panel (e) shows one representative row of the transform domain point spread function (PSF) for 3-fold uniformly undersampled data (solid, magenta line) and for 3-fold randomly undersampled data (dotted, purple line). Similarly, panel (d) shows the corresponding solutions obtained by solving the CS problem (2.33). As can be seen, the random sampling leads to a nearly perfect reconstruction, whereas the uniform sampling leads to aliasing artifacts.

2.7.2 Measurement Incoherence

Having discussed the signals to be recovered, the second pillar of CS is concerned with the questions what kinds of measurements allow for an exact reconstruction of the sparse signals at hand. To allow reconstruction even from undersampled measurements, the information about the sparse signal should be well spread out in the measurement domain. If this is fulfilled, it seems likely that sufficient information for reconstruction remains in the measured data even if only a small number of measurements are performed.

A useful tool to assess the suitability of measurement and sparsity transformation is their mutual coherence. For its definition, let us assume both $\mathbf{H}_\Gamma$ and $\mathbf{\Psi}^\dagger$ to contain an orthonormal basis in $\mathbb{K}^N$. Then the mutual coherence between these two bases is given by [50]

$$\mu(\mathbf{H}_\Gamma, \mathbf{\Psi}^\dagger) = \sqrt{N} \max_{1 \leq k,j \leq N} |\mathbf{h}_{k,:}^{\mathsf{H}} \boldsymbol{\psi}_{j,:}^\dagger|. \tag{2.30}$$

The mutual coherence takes on values in the interval $[1, \sqrt{N}]$ and it describes how correlated the two bases are. For the recovery of sparse signals it is desirable for the pair $\mathbf{H}_\Gamma$ and $\mathbf{\Psi}^\dagger$ to have low mutual coherence. If this is the case, any given basis function in $\mathbf{\Psi}$ will contribute measurably to a large number

of the samples in the measurement basis $\mathbf{H}_\Gamma$. Thus, for a signal that is sparse in the domain of $\mathbf{\Psi}$, information about the significant coefficients are likely to remain in the measured data even if only a subset of the measurements in $\mathbf{H}_\Gamma$ are performed. As a consequence of this, successful reconstruction can be achieved almost surely by picking measurements uniformly at random from the candidate set Γ. More details on this are collected in Theorem 2.2.

Another important tool for the analysis of the measurement and sparsity transformation are the restricted isometry constants of the composite transformation $\mathbf{H}_\Omega \mathbf{\Psi}^\dagger$. The latter describes the mapping from the sparsity domain into the undersampled measurement domain. The restricted isometry constant for this mapping is defined as the smallest number δ_S such that

$$(1 - \delta_S) \left\|\boldsymbol{\alpha}\right\|_2^2 \leq \left\|\mathbf{H}_\Omega \mathbf{\Psi}^\dagger \boldsymbol{\alpha}\right\|_2^2 \leq (1 + \delta_S) \left\|\boldsymbol{\alpha}\right\|_2^2 \tag{2.31}$$

for all S-sparse vectors $\boldsymbol{\alpha}$. When δ_S is small, the map $\mathbf{H}_\Omega \mathbf{\Psi}^\dagger$ approximately preserves the length of S-sparse signals. For CS this is beneficial because it implies that S-sparse vectors cannot be in the null space of $\mathbf{H}_\Omega \mathbf{\Psi}^\dagger$. Thus, no important information about the sparse signal at hand is lost during the measurement. In turn this implies that one should be able to recover the sparse signal from the measured data. Restricted isometry constants are particularly useful, because they incorporate the particular sampling set used. Thus, they can be used to determine whether a particular sampling scheme might allow successful signal recovery or not. A serious drawback is that restricted isometry constants are hard to compute, because they essentially require computation of the extreme eigenvalues of all submatrices of $\mathbf{H}_\Omega \mathbf{\Psi}^\dagger$ with S columns or less. Nevertheless, restricted isometry constants prove useful in the derivation of recovery and stability results (see e.g. Theorem 2.3 and Theorem 2.4).

A related quantity that can be computed more easily is the transform domain PSF $\left(\mathbf{H}_\Omega \mathbf{\Psi}^\dagger\right)^{\mathsf{H}} \mathbf{H}_\Omega \mathbf{\Psi}^\dagger$. The latter is a mapping from the sparsity domain into itself and it provides an estimate for the coherence between the different columns of $\mathbf{H}_\Omega \mathbf{\Psi}^\dagger$. If the off-diagonal elements of the transform domain PSF are small, the sparsest signal compatible with the measured data is unique. Thus, the underlying signal can be recovered by a suitable reconstruction method. On the other hand, large off-diagonal values indicate that multiple sparse signals might lead to the same measured data when applying the generalized measurement $\mathbf{H}_\Omega \mathbf{\Psi}^\dagger$. In this case, one cannot hope to accurately reconstruct the underlying signal by searching for a sparse solution that is consistent with the measured data.

To illustrate these sampling concepts, we briefly revisit the example in Figure 2.4. For the given combination of Fourier basis and canonical basis, the mutual coherence takes on its minimum value. Thus, the sparsity and measurement bases are maximally incoherent, which makes them suitable for CS. The example further illustrates two sampling schemes. On one hand, we consider a uniform sampling scheme were every third row of $\mathbf{H}_\Gamma$ is included in the undersampled measurement operator $\mathbf{H}_\Omega$. On the other hand, we consider a pseudo-random sampling scheme, where the same number of measurements were chosen uniformly at random. According to our previous discussion, one would expect the random sampling scheme to retain sufficient information about sparse signals to allow their reconstruction. This

is also reflected in the transform domain PSF, of which one representative row is shown in panel (e) of Figure 2.4. For the random sampling scheme, the transform domain PSF (dotted, purple line) contains a large peak at the diagonal matrix element and smaller noise-like values at the off-diagonal elements. It should thus be possible to recover the underlying signal, by searching for the sparsest solution consistent with the measured data. In contrast, the uniform sampling scheme has a transform domain PSF (solid, magenta line) with a main peak at the diagonal matrix element and 2 large peaks at off-diagonal elements. This indicates that a given measurement can be generated by multiple sparse signals, thus preventing a correct reconstruction of the underlying signal.

2.7.3 Signal Reconstruction

Based on the just discussed pillars, CS aims to reconstruct signals from undersampled measurement by searching for the sparsest possible solution that is compatible with the measured data. Mathematically, finding the sparsest solution to a given linear system is equivalent to finding the solution with minimal ℓ_0-norm. As a matter of fact, performing constrained ℓ_0-minimization is an NP-hard problem and thus often infeasible. For this reason, the ℓ_0-norm is replaced by the ℓ_1-norm, which is the convex relaxation of the former and which is known to be sparsity promoting [51]. Following this approach, a maximally sparse solution consistent with the measured data can be found by solving the constrained ℓ_1-minimization problem

$$\underset{\boldsymbol{\alpha}}{\operatorname{argmin}} \|\boldsymbol{\alpha}\|_1 \quad \text{subj. to} \quad \mathbf{y} = \mathbf{H}_\Omega \boldsymbol{\Psi}^\dagger \boldsymbol{\alpha}. \tag{2.32}$$

Having reconstructed $\boldsymbol{\alpha}$, the actual signal can then be obtained by computing $\mathbf{f} = \boldsymbol{\Psi}^\dagger \boldsymbol{\alpha}$.

A drawback of problem (2.32) is that it completely neglects the presence of measurement noise. This can be problematic, because measurement noise can easily cause the system $\mathbf{y}_\Omega = \mathbf{H}_\Omega \boldsymbol{\Psi}^\dagger \boldsymbol{\alpha}$ to no longer have a solution. In order to deal with cases where measurement noise cannot be neglected, one can consider the following relaxed version of the CS problem

$$\underset{\boldsymbol{\alpha}}{\operatorname{argmin}} \|\boldsymbol{\alpha}\|_1 \quad \text{subj. to} \quad \left\| \mathbf{y} - \mathbf{H}_\Omega \boldsymbol{\Psi}^\dagger \boldsymbol{\alpha} \right\|_2 \leq \varepsilon, \tag{2.33}$$

where ε is an upper bound on the expected measurement noise.

Coming back to the 1d-example in Figure 2.4, panel (d) shows CS reconstruction results for the random and the uniform sampling scheme described in subsection 2.7.2. Using the undersampled data, reconstruction was performed by solving problem (2.33) using the Split Bregman method [52]. As can be seen, the solution obtained using random sampling closely matches the underlying signal. On the other hand, the reconstruction obtained using uniform sampling displays clearly visible peaks at locations where the ground-truth signal is zero. These can be attributed to aliasing and they clearly reflect the structure of the corresponding transform domain PSF.

2.7.4 Recovery Guarantees and Stability Results

In the previous sections, we outlined how sparse signals can be recovered using CS. Moreover, we introduced several quantities, which help us assess whether a given measurement scheme allows an accurate reconstruction of sparse signals or not. A further requirement for a widespread adoption of CS is knowledge about the conditions under which an exact recovery of sparse signals can be guaranteed. For this reason, this section summarizes the main recovery guarantees and stability results derived in the development of CS.

To start of, we consider the noise-free setting, where sparse signals can be recovered by solving the constrained ℓ_1-minimization problem (2.32). If one uses measurements selected uniformly at random, the number of measurements required for an accurate reconstruction can be directly related to the sparsity of the signal and the mutual coherence of the sparsity and measurement bases via the following theorem [21]

Theorem 2.2. *Let $\mathbf{f} \in \mathbb{C}^N$ be S-sparse in the domain of the transform $\mathbf{\Psi}$. Moreover, let $\Omega \subseteq \Gamma$ be a set of M measurement indices selected uniformly at random. Then, if*

$$M \geq C \cdot S \cdot \log(N/\delta) \cdot \mu^2(\mathbf{H}_\Gamma, \mathbf{\Psi})$$

for some constant $C > 0$, the solution to (2.32) is an exact reconstruction of $\mathbf{f}$ with probability $1 - \delta$.

This result is particularly useful, because it provides a bound for a very concrete class of measurements. As can be seen, the required number of samples to achieve an exact recovery decreases proportionally to the degree of sparsity S. On the other hand, it grows quadratically with the mutual coherence between measurement basis and sparsifying transform.

While Theorem 2.2 is a powerful result, it does not take into account the measurement noise $\boldsymbol{\eta}_\Omega$, which typically corrupts measurements. To deal with this case, we consider problem (2.33), which contains a relaxed form of the data consistency constraint. For this setting, stability results can be formulated using restricted isometry constants. Using this concept, the following theorem bounds the recovery error of the solution to the CS problem (2.33) for the case of noisy measurements [19].

Theorem 2.3. *Let S be such that the restricted isometry constants fulfill $\delta_{3S} + 3\delta_{4S} < 2$. Then for any S-sparse signal $\boldsymbol{\alpha}_0$ and any perturbation $\boldsymbol{\eta}_\Omega$ with $\|\boldsymbol{\eta}_\Omega\|_2 \leq \varepsilon$, the solution $\boldsymbol{\alpha}$ to (2.33) obeys*

$$\|\boldsymbol{\alpha} - \boldsymbol{\alpha}_0\|_2 \leq C_S \cdot \varepsilon,$$

where the constant C_S may only depend on δ_{4S}. For reasonably small values of δ_{4S}, C_S is well-behaved; e.g. $C_S \approx 8.82$ for $\delta_{4S} = 1/5$ and $C_S \approx 10.47$ for $\delta_{4S} = 1/4$.

This result establishes that the solution to the CS problem is indeed stable in the presence of measurement noise and it relates the reconstruction error to the restricted isometry constant δ_{4S} of $\mathbf{H}_\Omega \mathbf{\Psi}^\dagger$. In contrast to Theorem 2.2, Theorem 2.3 does not assume a uniform random sampling of the measurement basis functions in $\mathbf{H}$ but it holds for arbitrary sets Ω leading to sufficiently small restricted isometry constants. This opens up the door to a further optimization of the measurements. This aspect is quite important because a purely random sampling can be difficult to implement for some applications in practice.

Another important point is that many signals are only approximately sparse. In order to be of practical use, the solution to problem (2.33) should also provide a stable estimate for signals that only allow for a sparse approximation in the domain of $\mathbf{\Psi}$. This kind of stability can be established by means of the following theorem [19].

Theorem 2.4. *Let $\boldsymbol{\alpha}_0 \in \mathbb{C}^N$ and let $\boldsymbol{\alpha}_{0,S}$ be the S-sparse vector whose non-zero components correspond to the S largest components of $\boldsymbol{\alpha}_0$. Under the conditions of Theorem 2.3, the solution $\boldsymbol{\alpha}$ to problem (2.33) obeys*

$$\|\boldsymbol{\alpha} - \boldsymbol{\alpha}_0\|_2 \leq C_{1,S} \cdot \varepsilon + C_{2,S} \cdot \frac{\|\boldsymbol{\alpha}_0 - \boldsymbol{\alpha}_{0,S}\|_1}{\sqrt{S}}.$$

For reasonably small values of δ_{4S}, the constants in this bound are well-behaved; e.g. $C_{1,S} \approx 12.04$ and $C_{2,S} \approx 8.77$ for $\delta_{4S} = 1/5$.

Here the reconstruction error is bounded by one contribution due measurement noise and an additional one due to the error of the S-sparse approximation to the underlying signal. This result greatly enhances the range of possible applications, as is justifies the use of CS for signals that are only approximately sparse.

2.7.5 Variations and Extensions

The provided recovery guarantees and early demonstrations of its applicability, have triggered a widespread adoption of CS-based techniques in many areas of application including MRI and MPI. Naturally, this has also led to many variations and extensions, which aim to adapt CS to the particular challenges and intricacies of the given application.

To obtain an efficient optimization algorithm, a commonly seen modification is to reformulate the constrained CS problem (2.33) as an unconstrained optimization problem. To do so, we introduce a Lagrange multiplier to enforce the data-consistency constraint. Moreover, we perform a change of variables so that optimization is directly performed with respect to the signal $\mathbf{f}$. By doing so, one ends up with the optimization problem

$$\underset{\mathbf{f}}{\operatorname{argmin}} \ \|\mathbf{y} - \mathbf{H}_\Omega \mathbf{f}\|_2^2 + \lambda \|\mathbf{\Psi} \mathbf{f}\|_1 . \tag{2.34}$$

This unconstrained formulation also exposes the close relationship between CS and MAP estimation. Comparing (2.34) with the problem (2.28), one sees that the unconstrained formulation of the CS problem

corresponds to the MAP estimator for a Gaussian likelihood function and a Laplacian prior distribution. An additional advantage of the unconstrained formulation is that unconstrained optimization problems can be solved quite efficiently using a variety of (sub)gradient-based methods, such as the conjugate gradients (CG) method [53], the fast iterative shrinkage-thresholding algorithm (FISTA) [54] or splitting based methods such as the split Bregman method [52] and the primal dual hybrid gradient (PDHG) method [43, 55].

An important limitation of many works on CS theory is the restriction to orthonormal sparsifying transforms. This hinders the application of CS in cases where signals do not have a sparse representation in an orthonormal basis but can be sparsified by other transformations such as an overcomplete dictionary. A variation that is commonly applied in imaging is known as total variation regularization. Here the ℓ_1-norm of the image gradients is penalized in order to obtain a solution with only a small number of sharp image edges. To address aforementioned limitation, a lot of effort has gone into the application of CS with non-orthogonal sparsifying transformations [26, 56]. Similarly, a large number of works investigate how to find transformations providing a maximally sparse representation for a given class of signals [31, 32].

Another approach pursued by many applications is to combine CS with other forms of prior knowledge about the signals at hand. This can be done either by replacing the ℓ_1-minimization typically used in CS by another more suitable norm or by adding further terms to the cost function. For instance, images are known to be sparse both when applying a Wavelet transform and when computing image gradients. Thus, a popular approach in CS-based image reconstruction is to combine an ℓ_1-term of the Wavelet coefficients with other ℓ_1-terms containing the image gradients [26, 52, 57]. Other works combine CS with other forms of regularization. For instance, image-series are often modeled as low-rank matrices. In this case, image reconstruction can be enhanced by combining the sparsity-promoting ℓ_1-term with another term promoting a low rank of the underlying matrix representation [58–60].

2.8 Error Measures

When developing algorithms for tasks, such as image reconstruction, extensive testing is required in order to make sure that the developed method actually reproduces high quality approximations to the actual image. Such tests heavily rely on suitable measures, which allow to quantify the similarity between a known ground-truth image and some approximation to it. In the following, we summarize two measures that will be used throughout this thesis. To simplify notation, we let $\mathbf{x}$ denote the ground-truth image and $\mathbf{y}$ an approximation, which was obtained e.g. by image reconstruction.

A commonly used measure for the similarity between these signals is the normalized root mean squared deviation (NRMSD), which can be defined as

$$\mathrm{NRMSD}(\mathbf{x}, \mathbf{y}) = \frac{\|\mathbf{x} - \mathbf{y}\|_2}{\sqrt{N}\,\|\mathbf{x}\|_\infty}, \tag{2.35}$$

with N being the number of elements in $\mathbf{x}$. The NRMSD is a popular measure for the difference of signals. One of its characteristics is that it is based on the ℓ_2-distance between the two signals compared. This structure implies, that the NRMSD performs a component-wise comparison of the signals and then aggregates the individual errors.

When used for the comparison of images, the component-wise comparison of the NRMSD is often criticized because the visually perceived similarity between images is not always captured well by a pixel-wise comparison. This can be explained by the observation that the perceived similarity between images depends strongly on the image structure, e.g. on the presence and location of edges and the resulting textures. For this reason, we also make use of the structural similarity index (SSIM), which aims to provide a measure for the perceived similarity of two images [61].

To compute the SSIM, we decompose the d-dimensional image into patches of size $P \times ... \times P$. Hence, let $\mathbf{x}^{\mathbf{r}} \in \mathbb{R}_+^{P^d}$ and $\mathbf{y}^{\mathbf{r}} \in \mathbb{R}_+^{P^d}$ denote the vector representation of the corresponding patches that are centered at the location $\mathbf{r}$. For these patches, let μ_x and $\sigma_x^2 = \frac{1}{P^d-1}\sum_{i=1}^{P^d}(x_i^{\mathbf{r}} - \mu_x)^2$ denote the mean and estimated variance of $\mathbf{x}^{\mathbf{r}}$. Similarly, we let μ_y and σ_y^2 denote the corresponding quantities for $\mathbf{y}^{\mathbf{r}}$ and we let $\sigma_{xy} = \frac{1}{P^d-1}\sum_{i=1}^{P^d}(x_i^{\mathbf{r}} - \mu_x)(y_i^{\mathbf{r}} - \mu_y)$ denote the covariance of $\mathbf{x}^{\mathbf{r}}$ and $\mathbf{y}^{\mathbf{r}}$. Using these quantities the SSIM of the patches is defined as

$$\mathrm{SSIM}(\mathbf{x}^{\mathbf{r}}, \mathbf{y}^{\mathbf{r}}) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}, \tag{2.36}$$

where $c_1 = (0.01L)^2$ and $c_2 = (0.03L)^2$ are used to stabilize the division with L denoting the dynamic range of the images. By construction, (2.36) yields a value in the range $[-1, 1]$ and a value of 1 implies that the patches $\mathbf{x}^{\mathbf{r}}$ and $\mathbf{y}^{\mathbf{r}}$ are identical [62]. Based on this patch-wise SSIM, the SSIM value for larger images can be obtained by computing the average SSIM over all image patches.

An aspect to keep in mind when using the SSIM is that the latter was originally developed for real-valued, non-negative images. When dealing with complex-valued images, we compute the SSIM separately for the real and imaginary parts and take the average of these two contributions. Moreover, we add a constant offset to all data prior to the SSIM calculation. This offset is chosen such that both real- and imaginary parts of the data become non-negative.

When aiming to compare sets of multiple images $\{\mathbf{x}_k\}_{k=1,...,K}$ and $\{\mathbf{y}_k\}_{k=1,...,K}$, we use the mean NRMSD

$$\overline{\mathrm{NRMSD}}(\{\mathbf{x}_k\}, \{\mathbf{y}_k\}) = \frac{1}{K}\sum_{k=1}^{K}\mathrm{NRMSD}(\mathbf{x}_k, \mathbf{y}_k). \tag{2.37}$$

In the same way, we compute the mean SSIM and denote it by $\overline{\mathrm{SSIM}}$.

3 Physical Background

As their name suggests, both MRI and MPI use magnetic fields in order to encode the distribution of the respective signal generating particles. To facilitate their description, this chapter reviews the basic principles describing electromagnetic fields and their interaction with matter.

3.1 Charges, Currents and Electromagnetic Fields

The central quantities in the classical theory of electromagnetism are particles with an electric charge. Such particles can be characterized using the charge density ρ, which describes their distribution, and the current density $\mathbf{j}$, which describes their motion state. The observable phenomena of electromagnetism arise as electric charges interact with each other. These interactions are mediated by the electric field $\mathbf{E}$ and the magnetic field $\mathbf{H}$, which are generated by the charged particles.

When placing dielectric materials in electric fields, one observes that the internal charges of the dielectric rearrange and form microscopic dipole moments. Macroscopically, these give rise to a polarization field, which opposes the external electric field. To describe such effect, without explicitly having to take into account all microscopic charges, one can use the displacement field

$$\mathbf{D} = \varepsilon_0 \mathbf{E} + \mathbf{P}. \tag{3.1}$$

Here $\varepsilon_0 \approx 8.854 \cdot 10^{-12}\,\mathrm{F\,m^{-1}}$ is the vacuum permittivity and $\mathbf{P}$ is the polarization field induced by the applied electric field. Similarly, when placed in a magnetic field, microscopic magnetic moments within a magnetic material can manifests themselves macroscopically in the form of a magnetization field. Often, these can be attributed to microscopic currents within the magnetic material. In order to describe such effects, without having to take into account all microscopic currents, we introduce the magnetic flux density

$$\mathbf{B} = \mu_0(\mathbf{H} + \mathbf{M}). \tag{3.2}$$

Here, $\mu_0 = 4\pi \cdot 10^{-7}\,\mathrm{H\,m^{-1}}$ is the vacuum permeability and $\mathbf{M}$ is the magnetization induced by the magnetic field.

To describe all electromagnetic phenomena in a unified way, one can use Maxwell's equations, which were proposed by J. C. Maxwell between 1861 and 1864 [63, 64], together with the material equations

(3.1) and (3.2). Taking into account the presence of dielectrics and magnetic materials, they can be written in the form of the following set of coupled partial differential equations:

$$\nabla \cdot \mathbf{D} = \rho_f \tag{3.3a}$$

$$\nabla \cdot \mathbf{B} = 0 \tag{3.3b}$$

$$\nabla \times \mathbf{E} = -\frac{\partial \mathbf{B}}{\partial t} \tag{3.3c}$$

$$\nabla \times \mathbf{H} = \mathbf{j}_f + \frac{\partial \mathbf{D}}{\partial t} \tag{3.3d}$$

Here ρ_f denotes the distribution of free charges and $\mathbf{j}_f$ denotes the corresponding free current density. Thus, we do not consider localized charges and currents bound to the dielectric but model their behavior using the material-specific quantities $\mathbf{D}$ and $\mathbf{B}$.

Conceptually, each of Maxwell's equations describes a fundamental observation, which can be verified through adequately designed experiments.

- (3.3a) states that free electric charges are a source of the displacement field in the sense that charges are associated with a positive flux of field lines emanating from the location of the charge. It is also known as Gauß' law.
- (3.3b) states that the net outflow of the magnetic induction through a closed surface is zero. Thus, there is no equivalent of an electric charge for the case of magnetism. It is also referred to as Gauß' law for magnetism
- (3.3c) describes how a change in the magnetic field induces a rotational electric field. This is also known as Faraday's law of induction.
- (3.3d) states that magnetic fields can be generated both by free currents and by a dynamically changing displacement field. It is also referred to as Amperes circuit law.

3.2 Particle-Field Interactions

In order to describe magnetic-field based imaging processes, one needs to understand how a given particle interacts with an applied magnetic field. In the classical theory of electrodynamics, these interactions are determined by the associated charge density $\rho(\mathbf{r})$ of the particle and the corresponding current density $\mathbf{j}(\mathbf{r})$.

To facilitate the description of these interactions, we note that the signal-generating particles in MRI and MPI are microscopically small. Thus, the charge- and current densities of the individual particles are localized in a small volume V_p with characteristic length scale that is typically well below 100 nm. On the other hand, the imaging experiment only aims to resolve the bulk macroscopic response of the particles to the applied magnetic fields. For instance, the typical size of an imaging voxel is on the order of 1 mm^3. In such a case, the field-matter-interactions can be well-described by the lowest-order multipole moments of the charge density and the corresponding current density [64].

As an example, we consider the interaction with an applied electric field. The latter can be well described in terms of the total charge

$$q = \int_V \rho(\mathbf{r})\,\mathrm{d}^3 r,$$

which corresponds to the zero$^{\text{th}}$-order moment of the charge distribution, and of its electric dipole moment

$$\mathbf{d} = \int_V \mathbf{r}\rho(\mathbf{r})\,\mathrm{d}^3 r,$$

which corresponds to the first-order moment. In the absence of magnetic fields, this first order approximation allows us to write the potential energy of the particles as

$$E_{PE}(\mathbf{r}) \approx qU(\mathbf{r}) - \mathbf{d}\cdot\mathbf{E}(\mathbf{r}). \tag{3.4}$$

Here $U(\mathbf{r})$ is a scalar potential, which fulfills $\mathbf{E}(\mathbf{r}) = -\nabla U(\mathbf{r})$.

When turning to the interaction with an applied magnetic field, one finds that the corresponding zero$^{\text{th}}$-order moment vanishes. This is a consequence of Eq. (3.3b), which postulates that there exists no analogue to the electric charge for the case of magnetic fields. Thus, the leading-order moment is the magnetic dipole moment

$$\boldsymbol{\mu} = \frac{1}{2}\int_V \mathbf{r}\times\mathbf{j}(\mathbf{r})\,\mathrm{d}^3 r. \tag{3.5}$$

In an applied magnetic field, the potential energy of a particle becomes

$$E_{PB} \approx -\boldsymbol{\mu}\cdot\mathbf{B}. \tag{3.6}$$

Moreover, the particle experiences a torque

$$\boldsymbol{\tau} = \boldsymbol{\mu}\times\mathbf{B}. \tag{3.7}$$

In contrast to the electric charge, which is a scalar quantity, both electric and magnetic dipole moments are vector-valued quantities. Thus, they are characterized by their magnitude and their spatial orientation. When brought into an external field, particles tend to align their dipole moments with the applied fields. As can be seen from Eqs. (3.4) and (3.6), this allows them to minimize their potential energy.

An important aspect specific to magnetic dipole moments is their close connection with angular momentum. To make this connection explicit, let us consider a collection of particles with mass m and charge q. Moreover, let $c(\mathbf{r})$ denote the spatially dependent particle density (i.e. concentration) and $\mathbf{v}(\mathbf{r})$ their velocity. In this case, Eq. (3.5) can be rewritten in terms of the angular momentum

$$\mathbf{L} = m\int_V \mathbf{r}\times c(\mathbf{r})\mathbf{v}(\mathbf{r})\,\mathrm{d}^3 r$$

as

$$\boldsymbol{\mu} = \frac{q}{2} \int_V \mathbf{r} \times c(\mathbf{r})\mathbf{v}(\mathbf{r}) \, \mathrm{d}^3 r = \frac{q}{2m}\mathbf{L} = \gamma \mathbf{L}. \tag{3.8}$$

Thus, the magnetic moment of a distribution of classical particles is proportional to its angular momentum with the proportionality constant given by the gyromagnetic ratio $\gamma = \frac{gq}{2m}$ and $g = 1$. For completeness, we note that the same relationship also holds for microscopic particles, such as electrons and nuclear spins. However, for these cases a quantum-mechanical description gives rise to different values for the constant g.

When performing biomedical imaging experiments, one is typically not interested in the dipole moment of single particles. Instead one considers the ensemble of all particles inside small volumes of interest ΔV. In order to macroscopically describe such particle ensembles, we use the polarization $\mathbf{P}$ and the magnetization $\mathbf{M}$. The polarization can be defined as the density of electric dipole moments in the sample volume

$$\mathbf{P} = \frac{1}{\Delta V} \sum_{i=1}^{N_p} \mathbf{d}_i, \tag{3.9}$$

where N_p denotes the number of particles in the respective voxel and the $\mathbf{d}_i$ denote the single dipole moments. In complete analogy, the magnetization is defined as the density of magnetic moments within the sample volume

$$\mathbf{M} = \frac{1}{\Delta V} \sum_{i=1}^{N_p} \boldsymbol{\mu}_i. \tag{3.10}$$

For the imaging modalities considered in this work, the magnetization turns out to be a useful quantity, because it is directly related to the concentration of the signal-generating particles. To formalize this, we now consider a collection of N_S different species, each of which containing particles with the same properties. In particular we make the assumption that the magnetic dipole vector has the same magnitude μ_ν for all particles of a given species ν. Furthermore, we assume that the orientation of particles can be described by a probability density function $f_\nu(\vartheta, \varphi)$, where ϑ denotes the polar angle and φ denotes the azimuthal angle in spherical coordinates. Under these assumptions the mean magnetic moment of a particle of species ν can be written as an expectation value over the distribution of particle orientations

$$\bar{\boldsymbol{\mu}}_\nu = \mu_\nu \int_0^\pi \mathrm{d}\vartheta \int_0^{2\pi} \mathrm{d}\varphi \begin{pmatrix} \cos\varphi \sin\vartheta \\ \sin\varphi \sin\vartheta \\ \cos\vartheta \end{pmatrix} f_\nu(\vartheta, \varphi).$$

As a result, the magnetization generated by species ν can be written as

$$\mathbf{M}_\nu = \frac{1}{\Delta V} \sum_{i=1}^{n_\nu} \boldsymbol{\mu}_{\nu,i} = c_\nu \bar{\boldsymbol{\mu}}_\nu, \tag{3.11}$$

where $\boldsymbol{\mu}_{\nu,i}$ denotes the magnetic moment of the i^{th} particle of species ν, and n_ν denotes the number of particles contained in the species. Moreover, $c_\nu = \frac{1}{\Delta V} n_\nu$ denotes the corresponding particle concentration.

In a similar way, the total magnetization generated by all species can be expressed in terms of the mean magnetic moment per particle, which is defined as

$$\bar{\boldsymbol{\mu}} = \frac{1}{N} \sum_{\nu=1}^{N_S} n_\nu \bar{\boldsymbol{\mu}}_\nu \tag{3.12}$$

with $N = \sum_{\nu=1}^{N_S} n_\nu$ denoting the total number of particles. With this at hand, the total magnetization can written as

$$\mathbf{M} = \frac{1}{\Delta V} \sum_{\nu=1}^{N_S} \sum_{i=1}^{n_\nu} \boldsymbol{\mu}_{\nu,i} = c\bar{\boldsymbol{\mu}}, \tag{3.13}$$

where $c = \frac{N}{V}$ is the overall particle concentration. Thus, we see that imaging of the concentration of particles can be performed, if one manages to encode the spatial distribution of the particle's magnetization. Such an encoding can be achieved by generating a spatially and temporally varying magnetization, while making the assumption that the underlying particle distribution remains unchanged during the measurement process.

3.3 Magnetic Dipoles

As sketched in the previous section, the magnetic dipole moment of particles can be used to describe how strong they are affected by an applied magnetic field. Vice versa, the interactions between applied fields and particles can be used to encode the distribution of the signal generating particles. Since the realization of such imaging schemes depends on the availability of particles with a suitable magnetic moment, this section discusses some examples of naturally occurring magnetic dipoles.

Magnetic dipoles arise in a magnitude of place in nature. The simplest example of a magnetic dipole is a current I that is constrained to a loop enclosing an area S. When applying (3.5) to the corresponding current density, one directly obtains a magnetic moment

$$\boldsymbol{\mu} = IS\mathbf{e}_S,$$

where $\mathbf{e}_S$ is a unit vector is normal to the surface enclosed by the current.

A second example of magnetic dipoles is caused by the bound electrons in an atom. Quantum-mechanically, the electrons are characterized using a discrete set of states each having a fixed energy and a fixed orbital angular momentum. According to Eq. (3.8), the associated electron current leads to an atomic magnetic moment. In practice, the orbital angular momentum of each electron is superimposed by another angular momentum which is termed spin. The spin is a non-classical property of electrons,

which can be derived using quantum field theory. Importantly, it cannot be linked to an associated electron current. The effectively observed magnetic moment is thus determined by the combination of orbital and spin angular momenta.

A related example are nuclear spins. These are particularly relevant for the field of MRI as they lie at the origin of the measured MRI signal. Experimentally, one finds that both protons and neutrons also possess an intrinsic angular momentum, which is called the nuclear spin. These nuclear spins lead to an additional contribution to the total magnetic moment of an atom. Nuclear spins and their use in MRI will be discussed further in chapter 4.

As a final example, we note that magnetic moments can also be formed by larger structures encompassing multiple atoms. Notably, interactions between atoms can lead to a coupling of the individual atom's magnetic moments. In materials, such as iron, this inter-atomic coupling leads to a ferromagnetic phase if the temperature falls below the Curie temperature T_C [65]. This phase is characterized by the formation of magnetic domains in which all magnetic moments are aligned parallel to each other. As a consequence, ferromagnetic materials, exhibit large magnetization values when placed in an external magnetic field. This ferromagnetic behavior of iron lies at the heart of the SPIOs, which are detected in MPI. Hence, this topic will be discussed further in chapter 5.

To finish this section, we note that almost all of the given examples are microscopic particles. To obtain a classical description of imaging experiments, it is important to properly choose the size of the sample volumes used to compute $\mathbf{M}$ for a given set of particles. Ideally, these volumes should be sufficiently large such that they contains a large number of particles and a macroscopic description is justified. On the other hand, they should be sufficiently small so that one can assume all particles within a volume to experience the same physical conditions. To simplify the discussion about such particle ensembles, one often describes them using an effective particle that behaves according to the ensemble average. For instance, in MRI, the effective magnetic dipole moment generated by an ensemble of nuclear spins is called an isochromat [66]. As a consequence, the signal in an imaging experiment can be explained classically in terms of multiple isochromats, without having to worry about quantum effects, such as particle statistics or the stochastic nature of the quantum-mechanical measurement process.

3.4 Measuring Magnetization Dynamics

In magnetic-field-based imaging experiments, the spatial distribution of particles is encoded by manipulating the magnetization in a controlled manner. Thus, a basic requirement for successful imaging is the ability to detect the dynamical changes of the magnetization within the field of view. Luckily, a mechanism for detecting such changes can be derived directly from Eq. (3.3c). The latter equation can be transformed into its integral form by performing integration over a surface S and applying Stokes' theorem

$$\oint_{\partial S} \mathbf{E}(\mathbf{r},t)\cdot \mathrm{d}\mathbf{s} = -\int_S \frac{\partial}{\partial t}\mathbf{B}(\mathbf{r},t)\cdot \mathrm{d}\mathbf{S}. \tag{3.14}$$

If one assumes the surface S to remain constant, the right hand side of this expression corresponds to the time derivative of the magnetic flux through the former. The left hand side can be identified with the potential along the path ∂S, which an electric probe would experience. To better understand this expression, let us consider a coil consisting of one infinitesimally thin winding, which encloses the surface S. According to (3.14), a change in the magnetic flux through S causes a corresponding voltage to be induced in the coil.

Next, let us consider a volumetric coil, consisting of multiple windings, covering the volume V. To determine the induced voltage, the electric field $\mathbf{E}$ needs to be integrated along all windings of the coil. To compute the corresponding contributions due to the magnetic flux, the surface integral in (3.14) needs to be replaced by a volume integral. By doing this, one obtains the following expression for the induced voltage in the coil

$$u(t) = -\int_V \mathbf{p}(\mathbf{r}) \cdot \frac{\partial}{\partial t}\mathbf{B}(\mathbf{r}, t)\, \mathrm{d}^3 r. \tag{3.15}$$

In this expression, the quantity $\mathbf{p}(\mathbf{r})$ is known as the coil sensitivity. Its form is determined by the coil geometry, e.g. the arrangement of the coil windings and the coil orientation.

A consequence of the law of induction is that receive coils can be used to detect temporal changes of magnetization. This can be directly inferred from the definition of the magnetic flux density in Eq. (3.2). Specifically, the voltage induced in a receive coil by a temporally varying magnetization can be written as

$$u(t) = -\mu_0 \int_V \mathbf{p}(\mathbf{r}) \cdot \frac{\partial}{\partial t}\mathbf{M}(\mathbf{r}, t)\, \mathrm{d}^3 r. \tag{3.16}$$

By combining Eq. (3.16) with Eq. (3.13), one can furthermore see that the voltage signal induced in a coil can be linearly related to the concentration of the signal-generating particles. This relationship is interesting because both imaging modalities considered in this work, encode the underlying particle concentration by manipulating the associated magnetization. Based on this, we can conclude that the measurements in both imaging modalities can be described using linear models with respect to the particle concentration. Hence, the framework described in chapter 2 can be used for their analysis and image reconstruction.

4 Magnetic Resonance Imaging

This chapter provides an introduction to the field of MRI. We start with a discussion of the physical principles underlying MRI and use these to derive the commonly used MRI signal models. Based on these, we provide an introduction to the most commonly used image reconstruction techniques and discuss some challenges currently faced by MRI research.

4.1 Introduction

MRI is a tomographic imaging technique, which is routinely used in a wide range of clinical applications [67, 68]. MRI uses magnetic fields to determine the distribution of a specific type of nuclear spins inside the sample to be imaged. The most common form of MRI targets ^{1}H protons, which have a high abundance in the human body and can be mainly found in water molecules but also in lipids and certain proteins.

Being a magnetic field-based imaging modality, MRI does not require the use of ionizing radiation. This makes it an attractive alternative to other modalities such as CT, where the dangers of radiation exposure are a non-negligible risk to be taken into account. A distinguishing feature of MRI is its excellent soft-tissue contrast, which is fundamentally caused by its sensitivity to ^{1}H protons. Beyond that, MRI offers a wide range of imaging sequences which permit to generate different tissue contrasts. This versatility makes MRI an attractive tool for clinical imaging applications. As a consequence, MRI is not only used for anatomical imaging, but also for the mapping of various physiological parameters such as blood flow [4], water diffusion [5] or the blood oxygenation in the human brain [6, 7].

Despite its many advantages, the applicability of MRI is often limited due to its relatively long acquisition times and high costs. These not only impose a limit on the achievable temporal resolution but also on the achievable spatial resolution. In recent years, these issues have been partly alleviated by the introduction of sophisticated image reconstruction methods, which allow image reconstruction from significantly fewer measurements than traditionally required [26, 27, 69–71].

4.2 Nuclear Spins

The central quantity of interest in MRI is the distribution of nuclear spins, which were introduced in section 3.3. In particular, the ^{1}H-atom contains a single proton and no neutrons. As a consequence, the resulting nuclear spin is unpaired, which results in a non-zero nuclear magnetic moment. Moreover, the ^{1}H-atom has a high abundance in the human body, which makes it a perfect target for MRI.

Similar to the electron spin, the nuclear spin is an intrinsic property of the particles in the atomic nucleus. Moreover, the description of nuclear spin dynamics must be based on quantum mechanics because there is no classical explanation for the nuclear spin [72]. Following the principles of quantum mechanics, the nuclear spin state of the ^{1}H-atom in a static magnetic field can be described as a superposition of two eigenstates $|\uparrow\rangle$ and $|\downarrow\rangle$. Here, the first state corresponds to the nuclear spin being aligned parallel and the second corresponds to the spin being aligned antiparallel to the magnetic field. The corresponding magnetic dipole moment can be expressed in analogy to the classical result in Eq. (3.8) as

$$\boldsymbol{\mu}_S = g\frac{e\hbar}{2m_p}\mathbf{S}, \tag{4.1}$$

with $g \approx 5.585$, the proton mass $m_p \approx 1.673 \cdot 10^{-27}\,\mathrm{kg}$ and the reduced Planck constant $\hbar \approx 1.055 \cdot 10^{-34}\,\mathrm{J\,s}$. The associated nuclear spin is $\mathbf{S} = \frac{\hbar}{2}\frac{\mathbf{B}}{|\mathbf{B}|}$ for the $|\uparrow\rangle$-state and $\mathbf{S} = -\frac{\hbar}{2}\frac{\mathbf{B}}{|\mathbf{B}|}$ for the $|\downarrow\rangle$-state.

By inserting Eq. (4.1) into Eq. (3.6), one finds that both eigenstates are separated by an energy barrier of $\Delta E = \frac{ge\hbar}{2m_p}|\mathbf{B}|$. As a result, nuclear spins preferably occupy the $|\uparrow\rangle$-state. At $T = 0$, this would imply a large magnetization as all magnetic moments in the given volume add up constructively. At non-zero temperatures, the situation changes because thermal fluctuations lead to a population of the $|\downarrow\rangle$-state. While the magnetization remains aligned parallel to the applied field, its magnitude is reduced due to the cancellation of magnetic moments with opposite orientation. To illustrate the effect of thermal fluctuations, let us consider a magnetic flux density of $B = 1\,\mathrm{T}$ and a temperature of $T = 300\,\mathrm{K}$. Assuming the nuclear spins to follow a Boltzmann distribution, one obtains a relative population difference between the $|\uparrow\rangle$ and the $|\downarrow\rangle$ state of $\approx 3.4 \times 10^{-6}$. Thus, only a fraction of 3.4×10^{-6} of all nuclear spins effectively contribute to the measurable magnetization at room temperature.

Finally, we note that the population of the $|\uparrow\rangle$ and the $|\downarrow\rangle$-states can be manipulated, by exposing the given nuclear-spin ensemble to radio-frequency pulses whose energy corresponds to the energy gap between both states. If this resonance condition is fulfilled, photons from the pulse can transfer their energy to nuclear spins thus allowing them to transition from one state to the other. The corresponding resonance frequency is given by

$$\omega = \frac{\Delta E}{\hbar} = \frac{ge}{2m_p}|\mathbf{B}| = \gamma\,|\mathbf{B}|\,. \tag{4.2}$$

This frequency is also known as the Larmor frequency.

4.3 Signal Encoding Principles

As suggested by its name, MRI is based on a phenomenon called nuclear magnetic resonance. The central idea of this imaging principle is to resonantly excite the nuclear spins of the object of interest. The resulting spin dynamics lead to a macroscopic magnetization dynamics, which can be measured and used to spatially encode the given nuclear spin distribution.

In MRI signal excitation and spatial encoding is achieved by a combination of three types of magnetic fields:

1. A strong static and homogeneous field $\mathbf{B}_0 = B_0\mathbf{e}_z$, which we assume to be aligned with the z-axis. In thermal equilibrium, the nuclear spins preferably align parallel to this field.
2. A homogeneous radio-frequency field $\mathbf{B}_1^+(t)$, which is usually circularly polarized within the xy-plane. This field is used to drive the nuclear spin ensemble away from thermal equilibrium and to induce a measurable magnetization dynamics.
3. A temporally varying gradient field $\mathbf{B}_G(\mathbf{r}, t) = (\mathbf{G}(t) \cdot \mathbf{r})\mathbf{e}_z$ is used to provide a spatial encoding of the underlying magnetization distribution.

In the following subsections each of these components will be discussed in more detail, as we derive a mathematical description of MRI experiments.

4.3.1 Signal Excitation

At the beginning of typical MRI experiments, only the $\mathbf{B}_0$-field is non-zero. Assuming thermal equilibrium, this leads to a macroscopic magnetization due to the nuclear spins preferably aligning with $\mathbf{B}_0$ as described in section 4.2 and illustrated on the left-hand side of Figure 4.1. As a matter of fact, this equilibrium magnetization is static, which means that it does not generate a measurable signal when using the inductive measurements described in chapter 3.

To induce measurable magnetization dynamics, a $\mathbf{B}_1^+$-field is used. If the frequency of the applied field matches the resonance frequency of the nuclear spins, this induces a population transfer between the different nuclear spin states. In turn, this leads to a rotation of the magnetization away from the z-axis and towards the xy-plane. By applying a pulse with suitable strength and duration, the magnetization can be tipped away from the z-axis by a predefined flip angle α. This is illustrated in the middle panel of Figure 4.1. Note that the magnetization trajectory drawn there has a more complicated, spiral-like form. This form arises because the rotation due to the $\mathbf{B}_1^+$ is superimposed by a rotational magnetization dynamics, which will be described in subsection 4.3.2.

Based on the spins resonance condition, signal excitation can also be used to realize an initial spatial encoding. To provide such an encoding let us consider an excitation pulse with central frequency ω and bandwidth $\Delta\omega$. Such a field can resonantly excite all nuclear spins whose resonance frequency is

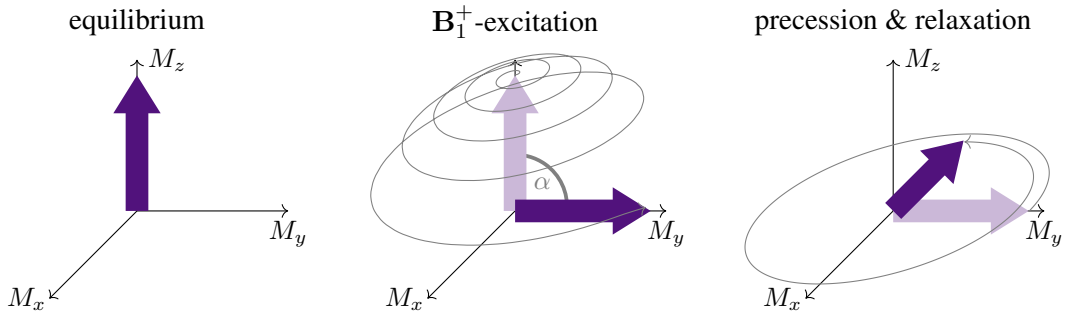


Figure 4.1: Magnetization dynamics in MRI experiments. In equilibrium the net magnetization is aligned with the applied $\mathbf{B}_0$-field (left). Application of a resonant $\mathbf{B}_1^+$-pulse tips the magnetization into the xy-plane (center). The tipped magnetization continues to precess around the z-axis. At the same time its magnitude decays due to relaxation (right).

contained within its bandwidth. To achieve a spatial encoding, the gradient fields can be used, since their presence leads to a spatially dependent resonance frequency

$$\omega(\mathbf{r}, t) = \gamma(B_0 + \mathbf{G}(t) \cdot \mathbf{r}). \tag{4.3}$$

In MRI, this mechanism is commonly used for selection of a 2d slice to be imaged. For a slice in the xy-plane this can be realized with a slice selection gradient of the form $\mathbf{G}_{ss} = G_{ss}\mathbf{e}_z$. If applied during signal excitation, only nuclear spins within a slab of a width $\Delta z \propto \frac{\Delta\omega}{G_{ss}}$ will be excited. Nuclear spins outside this slab will not be excited because their resonance frequency is not contained in the excitation pulse. In a completely analogous manner, slabs in the x- or y-direction can be obtained by gradients of the form $\mathbf{G}_{ss} = G_{ss}\mathbf{e}_x$ and $\mathbf{G}_{ss} = G_{ss}\mathbf{e}_y$.

4.3.2 Magnetization Dynamics in MRI

The magnetization dynamics in MRI can be described using the Bloch equations. These are a set of phenomenological equations of motions describing the dynamics of magnetization under the influence of a dynamically varying magnetic field [73]. They are given by

$$\frac{d\mathbf{M}(\mathbf{r}, t)}{dt} = \gamma\mathbf{M}(\mathbf{r}, t) \times \mathbf{B}(\mathbf{r}, t) - \begin{pmatrix} M_x(\mathbf{r}, t)/T_2 \\ M_y(\mathbf{r}, t)/T_2 \\ (M_z(\mathbf{r}, t) - M_0(\mathbf{r}))/T_1 \end{pmatrix}, \tag{4.4}$$

where T_1 and T_2 are relaxation times depending on the environment of the particles. Moreover, $M_0(\mathbf{r}) := |\mathbf{M}(\mathbf{r}, 0)|$ denotes the length of the equilibrium magnetization vector. In the limit $T_1, T_2 \to \infty$, the term on the right hand side becomes 0. The remaining terms simply correspond to the well known definition of torque as the temporal derivative of the angular momentum. To see this, note that $\frac{\mathbf{M}}{\gamma}$ corresponds to the angular momentum associated with $\mathbf{M}$, while $\mathbf{M} \times \mathbf{B}$ is the torque induced by the magnetic field.

In the absence of the $\mathbf{B}_1^+$-field, all remaining fields are aligned with the z-axis. If we further assume that the dynamics of $\mathbf{B}_G$ take place on a longer time scale than the magnetization dynamics, the solution to the Bloch equations without relaxation can be written as

$$\mathbf{M}(\mathbf{r},t) = \begin{pmatrix} |M_{xy}(\mathbf{r},0)| \cos\left(-\varphi(\mathbf{r},t)+\varphi_0\right) \\ |M_{xy}(\mathbf{r},0)| \sin\left(-\varphi(\mathbf{r},t)+\varphi_0\right) \\ M_z(\mathbf{r},0) \end{pmatrix}, \tag{4.5}$$

with the accumulated rotation angle $\varphi(\mathbf{r},t) = \gamma \int_0^t B(\mathbf{r},\tau)\,\mathrm{d}\tau$. From this, one can see that after signal excitation with a $\mathbf{B}_1^+$ field, the x- and y-components of the magnetization precess around the z-axis with the Larmor frequency $\omega(\mathbf{r},t) = \gamma B(\mathbf{r},t)$. The orientation of the magnetization at $t = 0$ is determined by the phase offset φ_0. An illustration of these magnetization dynamics is contained in the right-hand panel of Figure 4.1.

In practice, the described precession is perturbed by relaxation processes due to finite relaxation times T_1 and T_2. Taking into account these effects, the solution to the Bloch equations becomes

$$\mathbf{M}(\mathbf{r},t) = \begin{pmatrix} |M_{xy}(\mathbf{r},0)|\, \mathrm{e}^{-t/T_2(\mathbf{r})} \cos\left(-\varphi(\mathbf{r},t)+\varphi_0\right) \\ |M_{xy}(\mathbf{r},0)|\, \mathrm{e}^{-t/T_2(\mathbf{r})} \sin\left(-\varphi(\mathbf{r},t)+\varphi_0\right) \\ M_z(\mathbf{r},0)\mathrm{e}^{-t/T_1(\mathbf{r})} \end{pmatrix} + |M(\mathbf{r},0)| \begin{pmatrix} 0 \\ 0 \\ 1-\mathrm{e}^{-t/T_1(\mathbf{r})} \end{pmatrix}. \tag{4.6}$$

The relaxation mechanisms have two simple interpretations. On one hand, T_1- relaxation leads to a re-alignment of the magnetization vector with the applied $\mathbf{B}$-field, thus restoring longitudinal magnetization. This is why it is also referred to as longitudinal relaxation. This relaxation is mediated by interactions with nearby atoms and molecules. On the other hand, T_2-relaxation leads to an exponential decay of the magnetization stored in the transverse components M_x and M_y. For this reason it is also called transverse relaxation. This relaxation is caused by interactions between different nuclear spins.

In real MRI experiments, one often observes that the transverse magnetization components M_x and M_y decay faster than predicted by the T_2-values of the given material. This faster decay is caused by inhomogeneities of the main magnetic field $\mathbf{B_0}$. The latter leads to location-dependent changes of the precession frequency of the isochromats in an imaging voxel. In turn, the differences in precession frequency lead to a dephasing which explains the accelerated decay of M_x and M_y. In order to model this behavior, one typically retains the exponential relaxation behavior, but replaces T_2, by another constant T_2^* that incorporates both T_2-relaxation as well as the field-induced dephasing.

4.3.3 The MRI signal

Having obtained a model for the magnetization dynamics in MRI, we now aim to find a model for the voltage induced by it in a set of L receive coils. Such a model can be obtained by taking the time derivative of (4.6) and inserting it into (3.16). When computing the time-derivatives, one should note that

$M_z(\mathbf{r}, t)$ typically varies slowly compared to the precession frequency ω. Thus contributions due to M_z can be neglected and one obtains

$$u_l(t) = -\mu_0 \int_V p_{l,x} \frac{\partial M_x(\mathbf{r}, t)}{\partial t} + p_{l,y} \frac{\partial M_y(\mathbf{r}, t)}{\partial t} \, \mathrm{d}^3 r, \tag{4.7}$$

where the subscript l is used to index the different receive coils. The computation of the remaining partial derivatives can be further simplified by realizing that in most practical cases $\omega(\mathbf{r}, t) \gg 1/T_2^*(\mathbf{r})$. Thus, we can neglect terms involving the partial derivative of $\mathrm{e}^{-t/T_2^*(\mathbf{r})}$. By inserting the resulting time derivatives into (4.7), the induced voltage signal becomes

$$u_l(t) = -\mu_0 \int_V \omega(\mathbf{r}, t) \, |M_{xy}(\mathbf{r}, 0)| \, \mathrm{e}^{-t/T_2^*(\mathbf{r})} \tag{4.8}$$

$$\left\{ p_{l,x}(\mathbf{r}) \sin\left(- \int_0^t \omega(\mathbf{r}, \tau) \, \mathrm{d}\tau + \varphi_0(\mathbf{r}) \right) \right. \tag{4.9}$$

$$\left. - \, p_{l,y}(\mathbf{r}) \cos\left(- \int_0^t \omega(\mathbf{r}, \tau) \, \mathrm{d}\tau + \varphi_0(\mathbf{r}) \right) \right\} \mathrm{d}^3 r. \tag{4.10}$$

Here we made the additional assumption that the temporal changes of the Larmor frequency also happen on a slower time-scale than the magnetization dynamics and thus can be neglected.

To further simplify this expression, we rewrite the coil sensitivities using a polar decomposition

$$\mathbf{p}_l(\mathbf{r}) = \begin{pmatrix} |p_{l,xy}(\mathbf{r})| \cos\left(\vartheta(\mathbf{r})\right) \\ |p_{l,xy}(\mathbf{r})| \sin\left(\vartheta(\mathbf{r})\right) \\ p_{l,z}(\mathbf{r}) \end{pmatrix}.$$

Using this representation, the induced voltage can be rewritten as

$$\begin{aligned} u_l(t) = -\mu_0 \int_V \omega(\mathbf{r}, t) \, |p_{l,xy}(\mathbf{r})| \, |M_{xy}(\mathbf{r}, 0)| \, \mathrm{e}^{-t/T_2^*(\mathbf{r})} \\ \sin\left(- \int_0^t \omega(\mathbf{r}, \tau) \, \mathrm{d}\tau + \varphi_0(\mathbf{r}) - \vartheta(\mathbf{r}) \right) \, \mathrm{d}^3 r. \end{aligned} \tag{4.11}$$

This expression clearly shows that the voltage induced by the excited magnetization depends linearly on the transverse magnetization and that it features high-frequency oscillations, which are determined by the local Larmor frequency of the transverse magnetization. In MRI, the local Larmor frequency is mainly determined by the main magnetic field as the latter usually is a lot stronger than the applied gradient field. For this reason, it makes sense to write the Larmor frequency as

$$\omega(\mathbf{r}, t) = \omega_0 + \tilde{\omega}(\mathbf{r}, t). \tag{4.12}$$

Here $\omega_0 = \gamma B_0$ denotes the Larmor frequency caused by the B_0-field only and $\tilde{\omega}(\mathbf{r}, t)$ is a small frequency shift caused by the applied gradient field as well as imperfections of the B_0-field.

To facilitate the processing of the measured MRI data, the induced high frequency signal is moved into a low frequency band using signal demodulation. This is achieved by first multiplying the measured signal by a sinusoidal reference signal and then applying a low pass filter to remove the high frequency component of the signal [3]. If we assume a reference signal of the form $u_{\text{ref}}(t) = 2\cos(\omega_0 t)$, the product of both signals yields

$$\begin{aligned} u_l(t)u_{\text{ref}}(t) = &-\mu_0 \int_V \omega(\mathbf{r},t)\,|p_{l,xy}(\mathbf{r})|\,|M_{xy}(\mathbf{r},0)|\,\mathrm{e}^{-t/T_2^*(\mathbf{r})} \\ &\sin\left(-2\omega_0 - \int_0^t \tilde{\omega}(\mathbf{r},\tau)\,\mathrm{d}\tau + \varphi_0(\mathbf{r}) - \vartheta(\mathbf{r})\right)\Big)\,\mathrm{d}^3 r \\ &-\mu_0 \int_V \omega(\mathbf{r},t)\,|p_{l,xy}(\mathbf{r})|\,|M_{xy}(\mathbf{r},0)|\,\mathrm{e}^{-t/T_2^*(\mathbf{r})} \\ &\sin\left(-\int_0^t \tilde{\omega}(\mathbf{r},\tau)\,\mathrm{d}\tau + \varphi_0(\mathbf{r}) - \vartheta(\mathbf{r})\right)\,\mathrm{d}^3 r. \end{aligned}$$

After low pass filtering, the signal can be written as

$$\begin{aligned} u_l^{\text{psd,1}}(t) \approx -\mu_0\omega_0 \int_V &|p_{l,xy}(\mathbf{r})|\,|M_{xy}(\mathbf{r},0)|\,\mathrm{e}^{-t/T_2^*(\mathbf{r})} \\ &\sin\left(-\int_0^t \tilde{\omega}(\mathbf{r},\tau)\,\mathrm{d}\tau + \varphi_0(\mathbf{r}) - \vartheta(\mathbf{r})\right)\,\mathrm{d}^3 r, \end{aligned} \tag{4.13}$$

where we additionally made the approximation $\omega(\mathbf{r},t) \approx \omega_0$. Similarly, demodulation with a reference signal of the form $u_{\text{ref}}(t) = 2\sin(-\omega_0 t)$ yields

$$\begin{aligned} u_l^{\text{psd,2}}(t) \approx -\mu_0\omega_0 \int_V &|p_{l,xy}(\mathbf{r})|\,|M_{xy}(\mathbf{r},0)|\,\mathrm{e}^{-t/T_2^*(\mathbf{r})} \\ &\cos\left(-\int_0^t \tilde{\omega}(\mathbf{r},\tau)\,\mathrm{d}\tau + \varphi_0(\mathbf{r}) - \vartheta(\mathbf{r})\right)\,\mathrm{d}^3 r \end{aligned} \tag{4.14}$$

A drawback of this demodulation scheme is the symmetry of the demodulated signals. For instance, the signal $u_l^{\text{psd,2}}(t)$ could be generated both by the angular frequencies $\tilde{\omega}$ as well as $-\tilde{\omega}$ and an analogous relationship holds for $u_l^{\text{psd,1}}(t)$. To resolve these issues the processed MRI signal is formed by combining the result of both demodulations in a complex-valued representation

$$\begin{aligned} u_l^{\text{proc}}(t) &= u_l^{\text{psd,2}}(t) + i u_l^{\text{psd,1}}(t) \\ &\approx -\mu_0\omega_0 \int_V p_{l,xy}^*(\mathbf{r}) M_{xy}(\mathbf{r},0) \mathrm{e}^{-t/T_2^*(\mathbf{r})} \mathrm{e}^{-\mathrm{i}\int_0^t \tilde{\omega}(\mathbf{r},\tau)\,\mathrm{d}\tau}\,\mathrm{d}^3 r. \end{aligned} \tag{4.15}$$

Here we further simplified the notation by introducing the complex-valued transverse magnetization $M_{xy}(\mathbf{r},0) = |M_{xy}(\mathbf{r},0)|\,\mathrm{e}^{\mathrm{i}\varphi_0(\mathbf{r})}$ and the complex coil-sensitivity profile $p_{l,xy}(\mathbf{r}) = |p_{l,xy}(\mathbf{r})|\,e^{\mathrm{i}\vartheta(\mathbf{r})}$.

4.3.4 Spatial Encoding

To enable MRI with high in-plane resolution, the slice encoding scheme from subsection 4.3.1 is generally combined with another spatial encoding scheme known as k-space encoding. Similarly to slice encoding, k-space encoding is based on manipulating the precession frequency of the magnetization using the gradient field $\mathbf{B}_G$. As will be discussed in this section, playing out gradient waveforms after signal excitation allows to encode the spatial frequencies contained in the magnetization image to be determined. MRI measurements are thus performed in the spatial frequency domain. Inspired by physics, the latter is commonly referred to as k-space.

To derive the k-space encoding, we note that the spatially-dependent precession frequency can be written as

$$\omega(\mathbf{r}, t) = \gamma(B_0 + \mathbf{G}(t) \cdot \mathbf{r}) + \Delta\omega_0(\mathbf{r}). \tag{4.16}$$

Here the term $\Delta\omega_0(\mathbf{r}) = \gamma\Delta B_0(\mathbf{r})$ describes all the inhomogeneity of the main magnetic field $\mathbf{B}_0$ and we neglect inhomogeneity of the time-dependent gradient field. In practice, $\Delta B_0 \ll B_0$ and thus ΔB_0 is often neglected as well. However, there are multiple applications where the neglect of field inhomogeneities leads to artifacts in the reconstructed images. In most MRI scanners, field inhomogeneities arise from two contributions. On one hand, the field generated by the main magnet itself has a certain degree of inhomogeneity. On the other hand, the sample to be imaged usually has a non-uniform magnetic susceptibility. These susceptibility variations lead to an additional inhomogeneous contribution to the effective B_0-field. To minimize ΔB_0, one often uses a set of shim-coils. In a calibration step, the current in these coils is adjusted in order to minimize field inhomogeneities in the volume of interest.

Based on our expression for the resonance frequency, one can formulate the most commonly used model for the measured MRI signal, which we summarize in the following Lemma:

Lemma 4.1. *After excitation with a $\mathbf{B}_1^+$-pulse, the relationship between transverse magnetization and the measured MRI signal is given by*

$$u_l^{proc}(t) = \int_V p_{l,xy}^*(\mathbf{r}) M_{xy}(\mathbf{r}, 0) \mathrm{e}^{-t/T_2^*(\mathbf{r})} \mathrm{e}^{-\mathrm{i}\Delta\omega_0(\mathbf{r})t} \mathrm{e}^{-2\pi\mathrm{i}\mathbf{k}(t)\cdot\mathbf{r}} \,\mathrm{d}^3 r, \tag{4.17}$$

where the sampled k-space nodes are given by

$$\mathbf{k}(t) = \frac{1}{2\pi} \int_0^t \mathbf{G}(\tau) \,\mathrm{d}\tau. \tag{4.18}$$

If the length of the measurement interval is short compared to T_2 and to $\Delta\omega_0^{-1}$, the measured signal simplifies to the Fourier transform

$$u_l^{proc}(t) = \int_V p_{l,xy}^*(\mathbf{r}) M_{xy}(\mathbf{r}, 0) \mathrm{e}^{-t_E/T_2^*(\mathbf{r})} \mathrm{e}^{-\mathrm{i}\Delta\omega_0(\mathbf{r})t_E} \mathrm{e}^{-2\pi\mathrm{i}\mathbf{k}(t)\cdot\mathbf{r}} \,\mathrm{d}^3 r. \tag{4.19}$$

Algorithm 1: MRI measurement scheme

Input: k-space profiles $\{\mathbf{k}_\gamma\}_{\gamma=1}^{\Gamma}$
sampling points $\{t_j\}_{j=1}^{N_{\text{samp}}}$

1: **for** $\gamma = 1, \dots, \Gamma$ **do**
2: Transfer magnetization into the transverse plan by applying a suitable $\mathbf{B}_1^+$-pulse.
3: Apply the gradient waveform $\mathbf{G}_\gamma(t)$ to generate the corresponding k-space profile.
4: $\mathbf{u}_{l,\gamma} \leftarrow$ sample the resulting signal at the time points $\{t_j\}_{j=1}^{N_{\text{samp}}}$
5: Wait for the magnetization to relax back towards thermal equilibrium, before applying the next RF-pulse.
6: **end for**

Output: sampled MRI signal $\mathbf{u}_l = \left(\mathbf{u}_{l,1}^\mathsf{T} \dots \mathbf{u}_{l,\Gamma}^\mathsf{T}\right)^\mathsf{T}$

Proof: Eq. (4.17) is obtained by plugging Eqs. (4.16) and (4.18) into (4.15). If the measurement interval is sufficiently short, one can make the approximations $\mathrm{e}^{-t/T_2^*(\mathbf{r})} \approx \mathrm{e}^{-t_E/T_2^*(\mathbf{r})}$ and $\mathrm{e}^{-\mathrm{i}\Delta\omega_0(\mathbf{r})t} \approx \mathrm{e}^{-\mathrm{i}\Delta\omega_0(\mathbf{r})t_E}$, which directly yields (4.19). □

From the measurement model (4.19), one can see that MRI encodes a Fourier representation of the contrast-weighted transverse magnetization $M_{xy}(\mathbf{r}, 0)\mathrm{e}^{-t_E/T_2^*(\mathbf{r})}\mathrm{e}^{-\mathrm{i}\Delta\omega(\mathbf{r})t_E}$ weighted by the coil sensitivity profile $p_l(\mathbf{r})$. To formally describe this, we consider a sequence consisting of Γ excitations. Following each excitation, gradient waveforms are played out to sample an associated k-space profile

$$\mathbf{k}_\gamma : [0, t_R] \to \mathbb{R}^3, \quad t \mapsto \frac{1}{2\pi} \int_0^t \mathbf{G}_\gamma(\tau)\, \mathrm{d}\tau. \tag{4.20}$$

Here, the excitations are indexed by γ and t_R denotes the repetition time, i.e. the time between subsequent radio frequency (RF) excitations. The values of $\mathbf{k}_\gamma(t)$ have a simple interpretation, as they correspond to the spatial frequency component being sampled at at time t of the measurement.

When measuring the magnetization-induced signal, the latter is digitized using an analog-to-digital converter (ADC). Thus, after each excitation one obtains a discrete measurement signal $\mathbf{u}_{l,\gamma} \in \mathbb{C}^{N_{\text{samp}}}$, which corresponds to the Fourier transform of the magnetization image, evaluated at the k-space nodes $\{\mathbf{k}_\gamma(t_j)\}_{j=1,\dots,N_{\text{samp}}}$. For clarity, the resulting measurement scheme is summarized in Algorithm 1. An illustration of this scheme can also be found in Figure 4.2, which shows a common pulse sequence to realize a Cartesian sampling of k-space.

The full measurement signal can be conveniently represented by concatenating the individual profiles into a single vector

$$\mathbf{u}_l = \left(\mathbf{u}_{l,1}^\mathsf{T} \dots \mathbf{u}_{l,\Gamma}^\mathsf{T}\right)^\mathsf{T}.$$

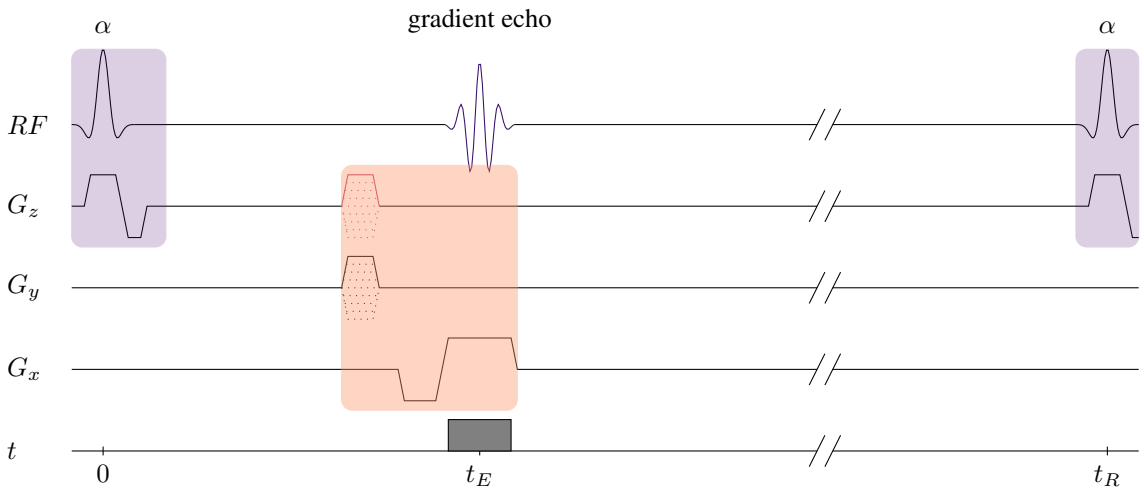


Figure 4.2: Basic pulse diagrams for a gradient echo sequence, which images a slice in the xy-plane. The slice selective excitation module is highlighted by a pale purple boxes. Similarly, the k-space encoding gradients are contained in the pale orange rectangle. Here G_x is used for frequency encoding and G_y is used for phase encoding. The gradient blips colored in pink denote the additional phase encoding gradients that would be employed in a 3d sequence. The time windows during which the ADC records the measured signal are indicated by a gray box on top of the time-line.

To simplify the notation, we use the same linear indexing for the sampling time and the k-space coordinates associated with each measurement. Thus, the sampling times are collected in a vector

$$\mathbf{t} = \left(t_1 \dots t_{N_{\text{samp}}} \dots t_1 \dots t_{N_{\text{samp}}}\right)^{\mathsf{T}} \in \mathbb{R}^K$$

and the corresponding k-space trajectory is given by the ordered set

$$\{\mathbf{k}_j = \mathbf{k}(t_j) \mid j = 1, \dots, J\}.$$

Here we let $J = N_{\text{samp}}\Gamma$ denote the total number of k-space samples.

4.3.5 k-space Sampling Schemes

Having established the Fourier measurement approach commonly used in MRI, we now turn to the task of designing appropriate sampling schemes, which allow for an accurate reconstruction of the underlying magnetization image. Classically, the design of such sampling schemes is guided by the Nyquist-Shannon sampling theorem, which is discussed in section 2.6. According to the NSST, an image with a given FOV and resolution can be reconstructed from a Cartesian set of Fourier measurements with k-space coordinates given by (2.29).

In MRI, the physical implementation of a Cartesian sampling scheme is based on the phase-encoding/ frequency-encoding scheme, which is illustrated in Figure 4.2. Here one of the gradients, e.g. G_y is

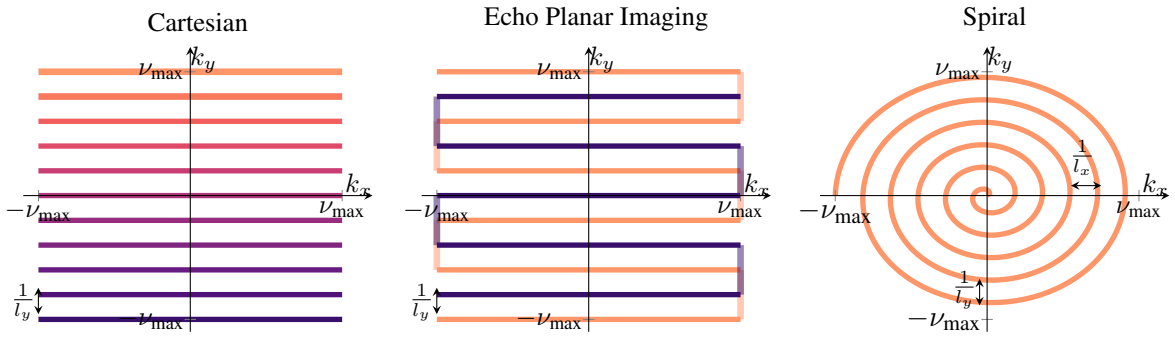


Figure 4.3: Illustration of some sampling trajectories commonly used in MRI. Shown are the commonly used Cartesian trajectory (left panel), an echo planar imaging trajectory consisting of two profiles (middle panel) and a spiral trajectory consisting of one single profile (right panel). The different profiles are plotted in different colors. The labels for the profile spacing illustrate the minimum distance between sampling points to fulfill the conditions of the NSST.

turned on for a brief period of time after the RF pulse has taken place. In this way the k_y coordinate is set to a fixed value corresponding to the time-integral of this short gradient pulse. This is referred to as phase encoding and the corresponding gradient is also called phase encoding gradient, which we abbreviate by G_{PE}. The remaining direction is encoded by setting the remaining gradient, e.g. G_x, to a fixed value during the time interval used to acquire the MRI signal. As a consequence, the value of k_x is continuously incremented between the subsequent samples acquired. This type of encoding is also called frequency encoding and the corresponding gradient is referred to as frequency encoding gradient, G_{FE}. By combining phase and frequency encoding as done in Figure 4.2, one line in k-space can be acquired after each excitation pulse. By exciting the system repeatedly, a full Cartesian dataset fulfilling the Nyquist-Shannon sampling theorem theorem can be obtained. The resulting k-space trajectory is illustrated in the left panel of Figure 4.3. Here the labels in the plot refer to the extend and spacing of the profiles required to achieve Nyquist sampling.

A limitation of the Cartesian sampling approach presented so far is that only one line in k-space is sampled after each RF excitation. This can lead to quite long measurement times as the number of phase encoding steps required for a full Cartesian dataset corresponds to the size of the reconstruction matrix along the phase encoding direction. Thus, the measurement time for a Cartesian 2d acquisition with an imaging matrix of size $N_x \times N_y$ is at least $\min(N_x, N_y)T_R$. To improve on this, gradient waveforms with longer readouts and more extended k-space coverage can be used. For instance, the echo planar imaging trajectory acquires multiple k-space lines after each excitation and thus reduces measurement time [74, 75]. This is illustrated in the middle panel of Figure 4.3. An even more efficient alternative is to sample k-space along spiral trajectories [76]. These allow to sample large portions of k-space with a small number of profiles, as illustrated in the right-hand panel of Figure 4.3. For some applications, these trajectories allow the acquisition of a full dataset using only a single RF excitation. Due to their short measurement times, these acquisition schemes also allow the visualization of fast dynamic processes, such as blood flow or the movement of the vocal cords [77, 78]. A drawback of these long-readout trajectories is that

inhomogeneities of the B_0-field cause deviations from the Fourier model (4.19). If not taken into account during reconstruction, these deviations cause characteristic artifacts, such as geometric distortions for the case of echo planar imaging or blurring for the case of spiral imaging [79–81].

4.3.6 3d Spatial Encoding

In order to encode the particle distribution in a 3d volume, one can combine a slice selective RF excitation with a frequency encoding and encode the remaining direction using phase encoding. This corresponds exactly to the encoding scheme illustrated in Figure 4.2. With this approach resolution in the slice-selection direction is limited by the spatial selectivity of the excitation pulse. A second limitation is that the number of signal-generating nuclear spins decreases together with the slice-thickness. Therefore signal-to-noise ratio decreases proportionally to the width of the excited slices.

Another approach to obtain a 3d encoding is to excite a wider slab of tissue and use a second phase encoding gradient to obtain a spatial encoding along the slice-selection direction. In Figure 4.2, these additional phase encoding gradients are highlighted in pink. This 3d encoding mechanism allows for a high-spatial resolution in all three dimensions without running into problems due to the excited slice profile. Moreover, a larger volume is excited, which leads to an overall larger signal and allows for a higher SNR. A drawback of this approach is that it can lead to increased imaging times due to the additional direction which needs to be phase encoded.

4.3.7 MRI Images & Image Contrasts

Having discussed the principles of MRI signal encoding, we now turn our attention to the question what kinds of information and image contrasts can be inferred from measured MRI data. From the signal equation (4.19) one can deduce that MRI provides information on the transverse magnetization in the form of an image

$$m(\mathbf{r}) = M_{xy}(\mathbf{r}, 0)\mathrm{e}^{-t_E/T_2^*(\mathbf{r})}\mathrm{e}^{-\mathrm{i}\Delta\omega_0(\mathbf{r})t_E}. \tag{4.21}$$

Thus, the MRI image is determined mainly by the transverse magnetization $M_{xy}(\mathbf{r}, 0)$ induced by RF-excitation. The image contrast is further influenced by the tissue specific T_2^* due to the term $\mathrm{e}^{-t_E/T_2^*(\mathbf{r})}$. The contribution of the T_2^* can be tuned by changing the echo time t_E. This is an important mechanism to manipulate image contrast in MRI. For instance, tissues with different T_2^*-values can be separated by choosing t_E to lie in the same range as the T_2^*-values under consideration.

An attractive feature of MRI is that it can not only generate T_2^*-weighted images but also other contrasts such as T_1- or T_2-weighted images. T_1-weighted contrast can be achieved by using a repetition time t_R in the same range is the as the T_1-relaxation times inside the imaged sample. As a consequence, the magnetization has no time to fully realign with the main magnetic field between excitations. In turn, this leads to a T_1-weighted image contrast. In medical applications, T_1-weighted contrast is often used to highlight fatty tissues. These appear bright in the corresponding images due to their short T_1 times.

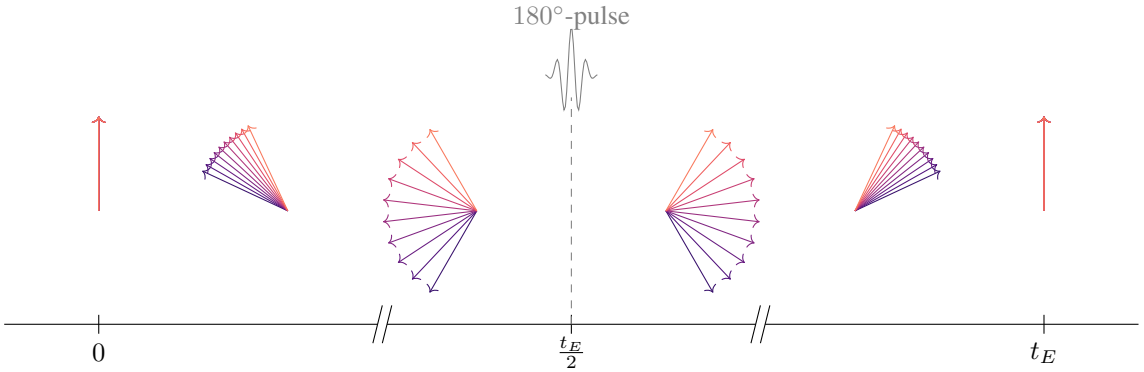


Figure 4.4: Illustration of field-induced spin dephasing. Different isochromats are shown in different colors. Local field inhomogeneity causes a dephasing of the isochromats. The application of a $180°$-pulses at $t = t_E/2$ rotates the magnetization by $180°$ around the x-axis. As time progresses, one observes a rephasing of the isochromats. Full rephasing is achieved at $t = t_E$.

Moreover, T_1-weighted imaging is often used with Gd-based contrast agents. These lead to a shortening of T_1 and thus also appear bright in the corresponding images.

The generation of T_2-weighted contrast requires a slightly modified pulse sequence, which mitigates the field-induced dephasing that characterizes T_2^*-relaxation. To better understand this, let us reconsider the MRI sequence shown in Figure 4.2. As can be seen, the frequency encoding gradient is set to a negative value directly before the signal acquisition starts. In this way, the first sample measured actually corresponds to a negative k_x-coordinate. As measurement time progresses, the $k_x(t)$ increases and a line with $k_x = 0$ in its center can be acquired. Another way to interpret this acquisition scheme can be obtained by considering the gradient induced dephasing of the transverse magnetization. Due to the negative frequency encoding gradient prior to data acquisition, the magnetization has a spatially dependent precession frequency which leads to an overall dephasing of the transverse magnetization. When the sign of the frequency encoding gradient is reversed, the spatial dependence of the precession frequency is exactly inverted. Thus, one can observe a rephasing of the magnetization. Full rephasing is achieved when the areas of the negative and the positive gradient lobe cancel out, at $t = t_E$. As time progresses further, one again observes a dephasing. In terms of signal intensities, the negative gradient lobe induces a decay of the measured signal. When the gradient polarity is reversed, the signal is recovered with a maximum occurring at the echo time. For this reason, this kind of signal is called gradient echo (GRE).

Mitigation of the field-induced dephasing can be achieved by spin echo sequences, such as the one illustrated in Figure 4.5. In spin echo sequences, an additional $180°$-pulse is applied at $t = T_E/2$. This pulse leads to a $180°$ rotation of the magnetization vector around the polarization axis of the pulse (e.g. the x-direction). After the $180°$-pulse, isochromats with a high precession frequency appear as if they had precessed slowly, whereas isochromats with a low precession frequency appear to have precessed fast. As the isochromats continue to precess, the field-induced dephasing will be inverted and full rephasing is

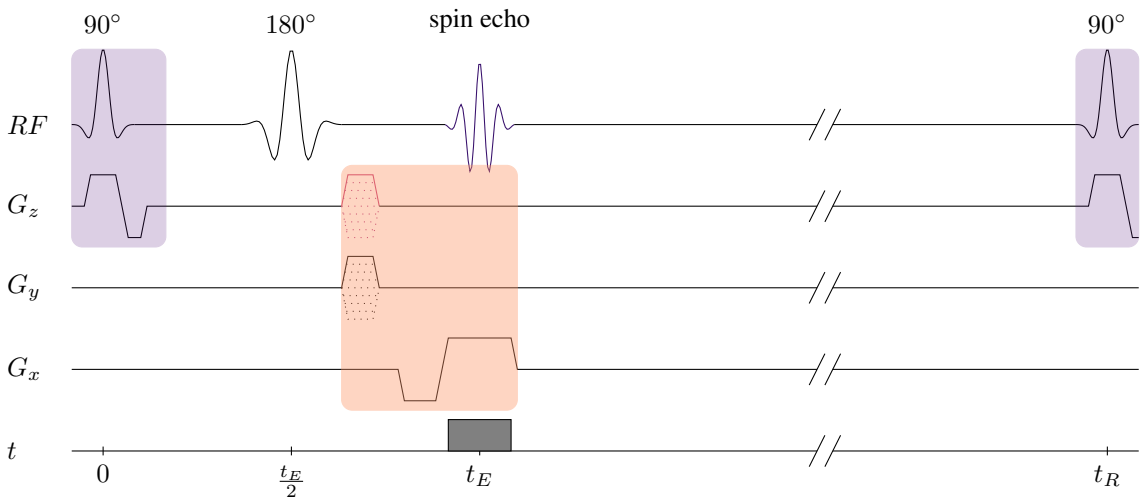


Figure 4.5: Basic pulse diagrams for a spin echo sequence, which images a slice in the xy-plane. The slice selective excitation module is highlighted by a pale purple boxes. Similarly, the k-space encoding gradients are contained in the pale orange rectangle. The gradient blips colored in pink denote the additional phase encoding gradients that would be employed in a 3d sequence. The time windows during which the ADC records the measured signal are indicated by a gray box on top of the time-line.

achieved at $t = t_E$. This rephasing mechanism is illustrated in Figure 4.4. In conclusion, the use of spin echo sequences, allows the generation of images with contrast depending on T_2, thus greatly increasing the versatility of MRI. In complete analogy to GRE sequences, the resulting image contrast can be tuned by selecting appropriate values of t_E.

4.4 MRI Signal Models

To connect MRI to the mathematical framework described in chapter 2, we now discuss the different signal models commonly used for image reconstruction along with their underlying assumptions. As outlined in sections 2.3 and 2.4, image reconstruction is usually performed using a discretized signal model of the form

$$\mathbf{u} = \mathbf{S}\mathbf{m} + \boldsymbol{\eta}, \tag{4.22}$$

where $\mathbf{u}$ contains the measured MRI signal and $\mathbf{m}$ is the discrete image to be reconstructed. Moreover, $\mathbf{S}$ is the MRI imaging operator relating both quantities and $\boldsymbol{\eta}$ contains the measurement noise. To make statistical inferences about $\mathbf{m}$, we thus need a model for the imaging operator as well as for the measurement noise.

4.4.1 Imaging Operators

To obtain an expression for the discretized MRI image, we apply the discretization (2.9) to Eq. (4.21), which leads to

$$\mathbf{m} = \left(M_{xy}(\mathbf{r}_i, 0) \mathrm{e}^{-t_E/T_2^*(\mathbf{r}_i)} \mathrm{e}^{-\mathrm{i}\Delta\omega_0(\mathbf{r}_i)t_E} \right)_{i=1,\ldots,N}. \tag{4.23}$$

The corresponding imaging operator is obtained by approximating the integral in Eq. (4.19) using a rectangular rule. Up to a constant, this leads to the following expression for the MRI signal induced in the l^{th}-receive coil

$$\mathbf{u}_l = \mathbf{F}\mathrm{diag}\left(\mathbf{p}_l\right)\mathbf{m}. \tag{4.24}$$

Here $\mathrm{diag}\left(\mathbf{p}_l\right)$ is a diagonal matrix describing the element-wise multiplication with the l^{th} coil sensitivity profile $\mathbf{p}_l = \left(p^*_{l,xy}(\mathbf{r}_i)\right)_{i=1,\ldots,N}$ and $\mathbf{F}$ is defined as

$$\mathbf{F} = \left(\frac{1}{\sqrt{N}} \mathrm{e}^{-2\pi\mathrm{i}\mathbf{k}_j\cdot\mathbf{r}_i} \mathrm{e}^{-(t_j-t_E)/T_2^*(\mathbf{r}_i)} \mathrm{e}^{-\mathrm{i}\Delta\omega_0(\mathbf{r}_i)(t_j-t_E)} \right)_{j=1,\ldots,J;i=1,\ldots,N}. \tag{4.25}$$

Eqs. (4.24) and (4.25) constitute the most general model used for standard MRI imaging. In particular, this model allows to take account system imperfections due to tissue relaxation and B_0-inhomogeneity.

To facilitate the reconstruction process, one often makes the assumption of negligible field inhomogeneity and relaxation, which is valid if the length of the measurement interval is sufficiently small. In this latter case (4.25) reduces to

$$\mathbf{F} = \left(\frac{1}{\sqrt{N}} \mathrm{e}^{-2\pi\mathrm{i}\mathbf{k}_j\cdot\mathbf{r}_i} \right)_{j=1,\ldots,J;i=1,\ldots,N}, \tag{4.26}$$

which has the form a discrete Fourier transform evaluated at the k-space trajectory nodes of the measurement.

If we consider a multi-channel acquisition, the measurement vector $\mathbf{u} = \left(\mathbf{u}_1^\mathsf{T} \ldots \mathbf{u}_L^\mathsf{T}\right)^\mathsf{T}$ is the concatenation of the individual coil data. The corresponding imaging operator can thus be written as

$$\mathbf{S} = \left(\left(\mathbf{F}\mathrm{diag}\left(\mathbf{p}_1\right)\right)^\mathsf{T} \ldots \left(\mathbf{F}\mathrm{diag}\left(\mathbf{p}_L\right)\right)^\mathsf{T} \right)^\mathsf{T} \tag{4.27}$$

This form of the MRI imaging operator is illustrated in Figure 4.6(a) for a simulated eight-channel acquisition of a digital brain phantom. The middle column clearly shows the spatial variations induced by the different coil sensitivities of the first three channels. These variations also propagate into the resulting k-space data, which is shown in the right-hand column.

An often used special case is that of a single-channel acquisition. In this case, the forward model only involves data acquired using a single receive channel. Moreover, one usually combines this model with the assumption of a homogeneous coil sensitivity profile, so that the imaging operator reduces to

$$\mathbf{S} = \mathbf{F}. \tag{4.28}$$

An advantage of this signal model is the simple structure of the imaging operator, which often simplifies image reconstruction. For this reason, the single-channel has often been used even in cases where multi-channel data was available. A drawback of this approach is that it discards any correlations between the data acquired with different receive channels, which can lead to sub-optimal reconstruction results.

4.4.2 Measurement Noise

When performing image reconstruction, the statistics of the measurement noise plays an important role, because different noise distributions lead to different statistical estimators for the underlying image. The choice of reconstruction methods can thus be guided by knowledge of the noise statistics. Beyond that, knowledge of the noise distribution allows us to provide estimates for the uncertainty of the reconstructed images.

In MRI, the predominant noise sources are thermal noise in the receive channel and in the sample itself. As a consequence, normally distributed measurement noise $\boldsymbol{\eta} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\boldsymbol{\eta}})$ is commonly assumed. For single-channel measurements, one can further assume a diagonal noise covariance matrix of the form $\boldsymbol{\Sigma}_{\boldsymbol{\eta}} = \sigma_{\eta}^2 \mathbf{1}$. When considering multiple receive coils, cross-talk between the different coils leads to a non-diagonal structure of the noise covariance matrix [82]. We note, however, that noise-whitening can be achieved by a suitable linear combination of the different coil measurements thus leading to a diagonal covariance matrix [83]. For this reason, we assume a diagonal noise covariance matrix with constant variances in the following.

4.5 Image Reconstruction

Having performed an MRI experiment, the next task is to obtain an estimate for the underlying magnetization image from the measured data. In MRI, there exist a multitude of reconstruction methods, which can be broadly categorized as direct and iterative reconstruction techniques. Direct reconstruction methods typically provide separate images for each receive channel used in the experiment. They are widely used due to their simplicity and the resulting high reconstruction speed. Iterative reconstruction methods, are computationally more demanding. On the other hand, these methods simplify the use of more sophisticated signal models as well as the incorporation of prior knowledge about the images at hand. Importantly, iterative reconstruction methods enable the adoption of advanced sampling techniques such as compressed sensing. The latter allows successful image reconstruction from greatly reduced datasets and can thus be used to reduce measurement time.

4.5.1 Direct Image Reconstruction

To derive a direct image reconstruction for MRI, we consider the MRI signal induced in a single-receive channel in the case of negligible B_0-inhomogeneity. In this case, the measured MRI signal is deter-

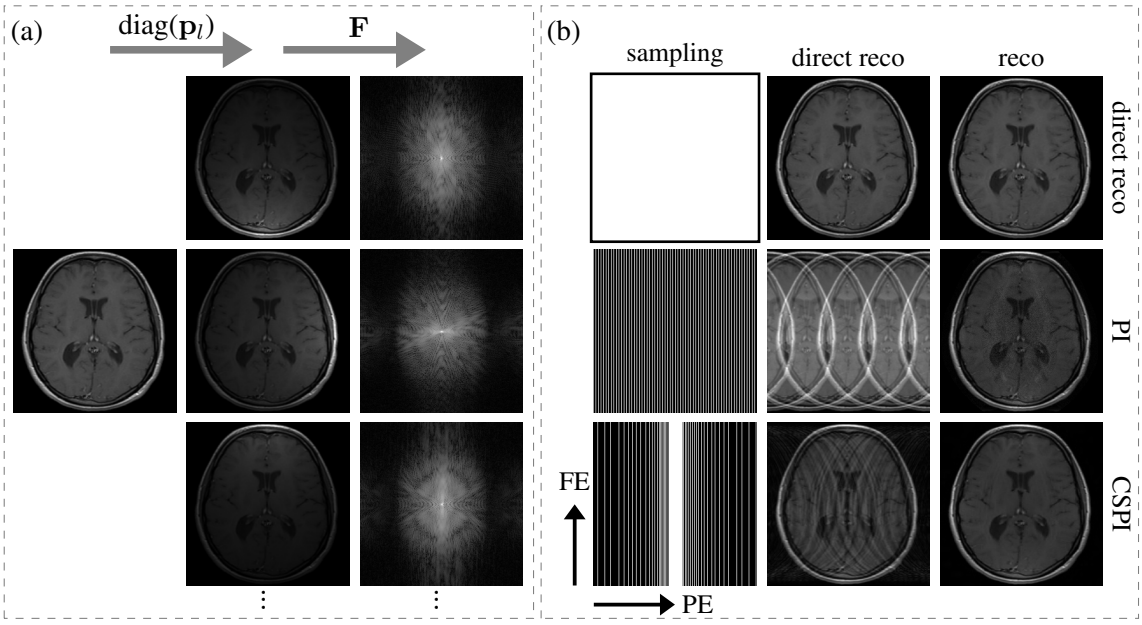


Figure 4.6: Illustration of the MRI image encoding operator (a) and the results of different reconstruction methods (b). The example considers an 8-channel acquisition of a digital brain phantom (left column). Applying the coil sensitivity maps leads to 8 weighted images of which the first three are shown in the middle column of panel (a). The modulus of the resulting k-space data for a Cartesian acquisition is shown in the right column (in log-scale). Panel (b) illustrates reconstruction results for a fully sampled direct reconstruction (top row), a PI reconstruction at 4-fold uniform subsampling (middle row) and a CSPI reconstruction from 4-fold undersampled data. The corresponding sampling schemes are shown in the left column. The corresponding phase encoding and frequency encoding directions are respectively labeled by PE and FE.

mined by Eq. (4.19), which can readily be inverted by applying an inverse Fourier transform. Thus, the underlying image can be written as

$$m(\mathbf{r}) = \int_V u^{\text{proc}}(\mathbf{k}) \mathrm{e}^{2\pi \mathrm{i} \mathbf{k}\cdot\mathbf{r}} \, \mathrm{d}^3 k. \tag{4.29}$$

If we now discretize this integral at the k-space nodes actually sampled in the imaging experiment, we can write the discrete image as

$$\mathbf{m}_{\text{direct}} \approx \mathbf{F}^{\mathsf{H}} \mathbf{W} \mathbf{u}, \tag{4.30}$$

where $\mathbf{W} = \operatorname{diag}\left((w_j)_{j=1,\ldots,J}\right)$ is a diagonal matrix containing quadrature weights appropriate for the given k-space nodes. Thus, image reconstruction can be performed by computing the adjoint DFT of the weighted k-space data $\mathbf{Wu}$. In MRI this direct reconstruction method is also referred to as gridding reconstruction.

The accuracy of the direct reconstruction strongly depends on the sampling density of k-space trajectory used for imaging. If the spacing between adjacent k-space nodes fulfills the conditions of the

NSST, successful image reconstruction can be expected. On the other hand, a too small sampling density leads to the situation that the image reconstruction problem becomes ill-posed. In this case, the direct reconstruction leads to artifacts in the reconstructed images, as is illustrated in Figure 4.6(b). The middle column shows results of the direct reconstruction method for a fully sampled acquisition (top row) and two different 4-fold undersampled sampling schemes (middle and bottom row). For the undersampled data, clear aliasing artifacts can be seen along the phase encoding direction.

To establish a connection to the statistical estimation methods introduced in chapter 2, we note that the direct reconstruction method is based on the approximation $\mathbf{F}^{\mathsf{H}}\mathbf{W}\mathbf{F} \approx \mathbf{1}_N$, which can be seen when combining (4.30) with the single-channel forward model. As a consequence, the result of the direct reconstruction method can also be written as

$$\mathbf{m}_{\text{direct}} = \mathbf{F}^{\mathsf{H}}\mathbf{W}\mathbf{u} \approx \left(\mathbf{F}^{\mathsf{H}}\mathbf{W}\mathbf{F}\right)^{-1}\mathbf{F}^{\mathsf{H}}\mathbf{W}\mathbf{u} \tag{4.31}$$

By comparing this with Eq. (2.18), we see that $\mathbf{m}_{\text{direct}}$ is an approximate solution to the weighted least squares problem

$$\underset{\mathbf{m}}{\operatorname{argmin}} \ \|\mathbf{u} - \mathbf{F}\mathbf{m}\|_{\mathbf{W}}^2 . \tag{4.32}$$

Thus, we can conclude that $\mathbf{m}_{\text{direct}}$ actually approximates the ML estimator, if we assume measurement noise of the form $\boldsymbol{\eta} \sim \mathcal{N}(\mathbf{0}, \sigma_\eta^2\mathbf{W})$.

A shortcoming of this method is that the assumed noise covariance matrix depends on the k-space trajectory used for sampling. In general, this does not match the actual noise statistics of the measurements. In turn this can lead to undesirable effects such as noise amplification. An exception worth highlighting is the sampling with a Cartesian trajectory. In this case the sampled k-space nodes are equidistant and thus the quadrature weights $(w_j)_{j=1,\dots,J}$ all have the same value. Moreover, $\mathbf{F}$ is a unitary operator in this case. Thus, the direct reconstruction reduces to

$$\mathbf{m}_{\text{direct}} = \mathbf{F}^{\mathsf{H}}\mathbf{u} = \left(\mathbf{F}^{\mathsf{H}}\mathbf{F}\right)^{-1}\mathbf{F}^{\mathsf{H}}\mathbf{u},$$

which is the solution to unweighted least-squares problem and corresponds to the ML estimator for the more realistic noise model discussed in subsection 4.4.2.

4.5.2 Parallel Imaging

A limitation of the direct reconstruction method is that images for each receive channel are reconstructed independently. Thus, it completely neglects all of the inter-channel correlations that are usually present in the measurement data detected using modern multi-channel receive coils [84]. Within the last 20 years, these drawbacks have led to the development of a large number of iterative reconstruction methods aiming to reconstruct images utilizing the information contained in all receive channels. This group of reconstruction techniques is commonly referred to as parallel imaging (PI) [27, 69, 70].

To illustrate the potential of PI, consider an MRI measurement performed with two receive coils. Moreover, let us assume that the first has a finite coil sensitivity only in one half of the FOV, whereas the second coil has a finite coil sensitivity only in the second half of the FOV. In this case, one half of the FOV can be reconstructed using the information from the first receive coil and the remaining half can be reconstructed using the data from the second receive coil. Since each channel is only used to reconstruct an image of half the total FOV, the number of phase encoding steps required for a successful image reconstruction can effectively be reduced by a factor of two, which implies a corresponding reduction in measurement time.

To describe PI image reconstruction, we consider a multi-channel acquisition with the measurement vector $\mathbf{u} = \left(\mathbf{u}_1^\mathsf{T} \dots \mathbf{u}_L^\mathsf{T}\right)^\mathsf{T}$. and the imaging operator given by (4.27). Many methods, including the popular sensitivity encoding (SENSE) method, use this forward model together with the Gaussian noise model discussed in subsection 4.4.2 [44]. Based on this statistical model an unbiased estimator for the image $\mathbf{m}$ can be obtained by computing the corresponding ML estimator, which is a solution to the problem

$$\underset{\mathbf{m}}{\operatorname{argmin}} \ \|\mathbf{u} - \mathbf{S}\mathbf{m}\|_2^2 \,. \tag{4.33}$$

As one can see, it takes into account the information from all receive channels, but does not impose any further constraints. As shown in subsection 2.5.1, the ML estimator can be obtained by computing the pseudo-inverse (2.20). A problem with this approach is, that it is not always feasible to evaluate the pseudo-inverse directly in MRI. This is mainly caused by the large size of the matrix $\mathbf{S}^\mathsf{H}\mathbf{S} \in \mathbb{C}^{N \times N}$, which makes numerical inversion impractical. To circumvent this issue, a common approach is the use of iterative methods such as the CG method.

Due to the more efficient use of information, PI allows for a significant reduction of the number of phase encoding steps compared to a fully sampled acquisition that fulfills the NSST. To achieve a robust reconstruction, most PI methods use a uniform sampling scheme where the acquired phase encoding lines are spaced evenly across k-space. The amount of undersampling to be achieved strongly depends on the number of receive coils and their precise arrangement. However, our previous example illustrates, that the undersampling factor should definitely not be larger than the number of receive channels used in the experiment. In practice, the achievable undersampling factor is often smaller due to effects such as channel-coupling between the receive coils and noise amplification [69].

An example of PI is shown in the middle row of Figure 4.6(b). Here only every fourth phase encoding line was sampled, as is illustrated in the left column. As a result, the direct reconstruction shows severe aliasing artifacts along the phase encoding direction. On the other hand, a SENSE-reconstruction using Eq. (4.33) clearly recovers the structure of the underlying image. Nevertheless, a more detailed look also reveals a certain amount of noise amplification, which manifests itself in the center of the image. This is a consequence of the ill-conditioning of the reconstruction problem, which results from the large undersampling factor used.

4.5.3 Compressed Sensing

A further group of techniques is based on the CS paradigm, which was introduced in section 2.7. A basic requirement for these techniques is knowledge of a transformation $\mathbf{\Psi}$, which maps the images to be reconstructed to a sparse signal. For MRI images several such transformations are known, one of the most popular examples being the Wavelet transform [26, 27]. Other approaches use transformations such as shearlets, framelets or image gradients [26, 30]. Moreover, a lot of effort has gone into the idea of learning tailored transforms which lead to maximally sparse representations for a given anatomy of interest [31, 32].

The second requirement is to adopt a sampling scheme with high incoherence. To achieve this, a straightforward approach is to use a pseudo-random selection of phase encodings, which is in line with the requirements of Theorem 2.2. When designing such sampling schemes it is beneficial to keep in mind that the measurements with the largest signal in an MRI datasets are usually contained in the center of k-space. This is to be expected because images typically contain many regions of constant intensities and a correspondingly large DC component of their spatial Fourier transform. Due to this property, it is common practice to perform CSMRI using variable density sampling patterns where phase encodings are still chosen pseudo-randomly but the center of k-space is sampled with an increased density [85–87].

Having performed CS measurements, image reconstruction is a direct application of the results in section 2.7. Thus, a CSPI image reconstruction, which combines CS and PI, can be formulated as the unconstrained optimization problem

$$\underset{\mathbf{m}}{\operatorname{argmin}} \ \|\mathbf{u} - \mathbf{S}\mathbf{m}\|_2^2 + \lambda \|\mathbf{\Psi}\mathbf{m}\|_1 \,. \tag{4.34}$$

As discussed previously, this can also be viewed as computing the MAP estimate for our given Gaussian noise model together with a Laplacian prior. As for PI, the solution of this optimization problem can be obtained using iterative methods, such as the nonlinear CG method, FISTA, alternating direction of multipliers method (ADMM) or the PDHG method [26, 43, 52, 54].

An example for CSPI image reconstruction is shown in the bottom row of Figure 4.6(b). Here, a pseudo-random variable density sampling scheme was used and we used a Wavelet transform as sparsifying transform. In the middle column, one sees that a direct reconstruction leads to aliasing artifacts, which, however, appear more noise-like than those in the uniform subsampling case. The CSPI reconstruction manages to accurately recover the underlying image. Notably, the noise-amplification, which is observable for the PI reconstruction, is clearly reduced for CSPI.

4.6 Challenges

In the last 15 years, MRI has seen tremendous advances through the introduction of advanced iterative reconstruction techniques. Methods like PI and CS now allow successful image reconstruction from measurements containing a lot fewer measurements than required by the NSST. In combination with

modern receive coils having up to 128 or more channels [88], acceleration factors on the order of 10 and higher can be achieved. Regardless of the advances made so far, there remain many challenges whose solution could lead to a further acceleration of MRI measurements, enable new applications and increase the diagnostic value of MRI.

In CSMRI, a lot of research aims at finding tailored sparsifying transforms, which might allow successful image reconstruction from an even smaller number of measurements. Besides traditional transforms, such as the Wavelet transform, data-driven approaches and machine learning techniques also receive a lot of attention in this context [89–91]. The latter techniques show great promise, because they allow a very flexible tailoring of reconstruction methods to the data and anatomy used in actual imaging experiments. On the other hand, this flexibility makes it hard to provide guarantees for the stability of reconstruction, especially when data imperfections are to be expected.

A further challenge in CSMRI is the design of sampling schemes allowing for accurate reconstruction from highly undersampled data. While stability results can be derived for uniform, pseudo-random sampling patterns, the latter patterns usually do not provide optimal performance [26, 33, 86]. In fact, it can be easily seen that the performance of a given sampling pattern depends on the structure of the images to be recovered. For this reason, current research also investigates data-driven techniques for the determination of sampling patterns that are tailored to a given anatomy at hand [33, 34]. This challenge will be investigated further in chapter 9.

Finally, we note that the acceleration of MRI measurements is enabled by the use of sophisticated and more complicated reconstruction methods. Thus, the bottleneck in image acquisition is effectively shifted from the imaging hardware to the software. As a consequence of this shift, fast and efficient reconstruction algorithms are a key requirement for the deployment of modern image reconstruction methods in practice. Based on these requirements, the development of software libraries that are both fast and accessible is an important catalyst for the development of new image reconstructions methods as well as their practical application. This is discussed in more detail in chapter 6.

A prominent example for the importance of fast algorithms is the use of non-Cartesian sampling trajectories in combination with CS or PI. While showing great promise for the further acceleration of MRI measurements, this combination of techniques is not employed very often, due to the challenges associated with image reconstruction. One major issue is that iterative reconstruction of non-Cartesian data is computationally more demanding than the reconstruction of Cartesian datasets. Another drawback is that the signal model (4.19) breaks down for trajectories with long readouts, such as spirals, leading to the appearance of artifacts. Correcting these artifacts is possible through the use of the more general signal model (4.17). As a matter of fact, this generalized signal model is even harder to evaluate numerically, thus leading to a further increased computational burden [42, 80, 81]. This example clearly illustrates how the development of faster algorithms can contribute to the shortening of measurement times in MRI. Approaches to accelerate image reconstruction in this setting are the subject of chapter 7.

5 Magnetic Particle Imaging

This chapter provides an introduction to the field of MPI. We start by describing the physical principles underlying MPI signal encoding. Based on these, we introduce the most commonly used MPI signal model along with a discussion of current image reconstruction techniques. Finally, we provide a discussion about some challenges faced by MPI, which currently hinder a more widespread use.

5.1 Introduction

Magnetic particle imaging is a relatively young imaging modality, which was invented by Bernhard Gleich and Jürgen Weizenecker in 2001 [9, 10]. The underlying idea of the method is to image the distribution of superparamagnetic iron nanoparticles after they have been administered to a patient. Following this idea, MPI can provide functional images containing information about the patients physiology. Importantly, MPI uses magnetic fields for the signal encoding. Thus, imaging can be performed without the use of ionizing radiation. This makes MPI particularly attractive when comparing it to other popular imaging techniques, such as PET, single photon emission computed tomography (SPECT) or CT. Furthermore MPI promises a high spatio-temporal resolution and a high sensitivity, which makes it a promising candidate for various clinical applications.

With regard to the sensitivity, we note that SPIOs can also be visualized using MRI, where they lead to an enhanced T_2^*-dephasing. A drawback is that the accelerated relaxation leads to a negative contrast, which makes it hard to quantify the SPIO concentration. In turn this can lead to problems when quantifying the lumen area in applications such as angiography [92]. With this in mind, its positive SPIO-contrast and the resulting high sensitivity make MPI an interesting alternative for angiography.

Turning to the applications of MPI we note that the distribution of tracer material can provide valuable information about the blood flow in the human body. For this reason, a lot of research has been performed on the MPI-based imaging of vascular diseases [11–13] and of stroke [16]. Furthermore, one can envision applications inspired by the molecular imaging routinely performed using the modalities MRI, PET and SPECT. The latter modalities allow the imaging of tracers, which have been designed to target specific sites in the body. For instance, ^{18}F deoxyglucose-based PET is a popular tool to characterize glycolysis in potentially cancerous cells [93] and Tl-201-based SPECT is commonly used to image myocardial perfusion [94]. The feasibility of cell-labeling-based imaging with MPI has been demonstrated

in [95]. Finally, MPI can be used for the tracking of surgical instruments during interventions [14, 15, 96].

Currently, MPI is in a preclinical stage. Scanners are readily available for small animal imaging. Moreover, recent works have resulted in scanners dedicated to the imaging of specific body parts such as the human head [97]. Besides the application-oriented research, a lot of the current research effort aims at solving technical issues such as the appearance of background signals [98, 99] and the long calibration times often encountered in MPI [28, 29].

5.2 Superparamagnetic Nanoparticles

Superparamagnetic iron oxide nanoparticles lie at the heart of MPI because their distribution is the quantity being measured by it. SPIOs consist of an iron core, which is responsible for its magnetic behavior and a magnetically neutral coating, which prevents agglomeration of the particles. Typically the core size of these particles lies in the range between 10 and 100 nm [100]. Due to their small sizes, SPIOs consist of a single magnetic domain. Nevertheless, the strong coupling between the spins composing the domain gives rise to a magnetic susceptibility that is significantly larger than that of paramagnets.

Unlike other ferromagnets, SPIOs show negligible hysteresis and have zero net magnetization when no external magnetic field is applied. To understand this effect, let us make the assumption that the magnetic moment of the SPIOs have two stable orientations, which are anti-parallel to each other. For most SPIOs this assumption indeed holds, due to the magnetic anisotropy of the particles. At finite temperature, transitions between the two states can occur whenever the thermal fluctuations are sufficiently strong to cross the energy barrier separating the two states. The time scale of this transitions can be characterized using the Néel relaxation time

$$\tau_N = \tau_0 \exp\left(\frac{KV}{k_B T}\right), \tag{5.1}$$

which depends mainly on the particles core volume V, its anisotropy energy density K and the surrounding temperature T [101]. Here k_B denotes the Boltzmann constant and τ_0 is a material specific time constant. To get a better feeling for τ_N, let us assume a particle with a core diameter of 10 nm, an anisotropy energy density of 1 kJ/m^3 and τ_0 of 1 ns. At 300 K this leads to a Néel relaxation time of $\tau_N \approx 2.75$ ns. This time is significantly smaller than the time needed for measurements of the magnetization, which typically is on the order of 40 µs on current MPI scanners. As a consequence, the net magnetization in the absence of a magnetic field is zero.

When the particles are suspended in a liquid, a second relaxation pathway opens. Instead of changing the internal spin state, a SPIO can also undergo a physical rotation. This process is called Brown relaxation and it can be characterized by the associated relaxation time

$$\tau_B = \tau_0 \frac{3\eta V_H}{k_B T}, \tag{5.2}$$

which is dependent on the viscosity of the surrounding medium η and the hydrodynamic volume of the particles V_H [101]. Under realistic conditions this Brown relaxation often takes place on a slower time scale than Néel relaxation [102].

5.2.1 Langevin Theory of Paramagnetism

To better understand the behavior of SPIOs in an imaging experiment, let us consider the magnetization generated by an ensemble of particles in a voxel of size ΔV. In order to take into account the heterogeneity of the particles used for imaging, we further consider a collection of N_S different species, each of which containing particles with the same properties such as core size. This setting corresponds exactly to the one discussed in section 3.2. Hence, the resulting magnetization can be described by Eq. (3.13).

When an external magnetic field $\mathbf{H}$ is applied, the described relaxation mechanisms cause the magnetization to align with the former. In the limit of sufficiently small relaxation times one can assume that the distribution of particles is always in thermal equilibrium. The evolution of the magnetization can than be described using the Langevin theory of paramagnetism [103]. In this model, each particle species leads to a magnetization of the form

$$\mathbf{M}_\nu(\mathbf{H}) = \frac{1}{\Delta V}\sum_{i=1}^{n_\nu} \boldsymbol{\mu}_{\nu,j} = \frac{n_\nu \left\|\boldsymbol{\mu}_\nu\right\|}{\Delta V}\mathcal{L}\left(\frac{\mu_0 \left\|\boldsymbol{\mu}_\nu\right\| \left\|\mathbf{H}\right\|}{k_B T}\right)\mathbf{e}_\mathbf{H}, \tag{5.3}$$

where

$$\mathcal{L}(x) = \begin{cases} \left(\coth(x) - \frac{1}{x}\right) & x \neq 0 \\ 0 & x = 0 \end{cases} \tag{5.4}$$

is the Langevin function and $\mathbf{e}_\mathbf{H}$ is a unit vector pointing in the direction of $\mathbf{H}$.

An interesting property of the Langevin model is that the magnetization is always aligned parallel to the applied magnetic field for all particle species. Thus, the mean magnetic moment per particle can be written as

$$\bar{\boldsymbol{\mu}}(\mathbf{H}) = \frac{1}{N}\sum_{\nu=1}^{N_S} n_\nu \left\|\boldsymbol{\mu}_\nu\right\| \mathcal{L}\left(\frac{\mu_0 \left\|\boldsymbol{\mu}_\nu\right\| \left\|\mathbf{H}\right\|}{k_B T}\right)\mathbf{e}_\mathbf{H}$$

and the total observed magnetization can be obtained by inserting this into Eq. (3.13)

$$\mathbf{M}(\mathbf{r},t) = c(\mathbf{r})\bar{\boldsymbol{\mu}}(\mathbf{H}(\mathbf{r},t)) = \frac{c(\mathbf{r})}{N}\sum_{\nu=1}^{N_S} n_\nu \left\|\boldsymbol{\mu}_\nu\right\| \mathcal{L}\left(\frac{\mu_0 \left\|\boldsymbol{\mu}_\nu\right\| \left\|\mathbf{H}(\mathbf{r},t)\right\|}{k_B T}\right)\mathbf{e}_\mathbf{H}. \tag{5.5}$$

An exemplary plot of the resulting magnetization curve for a single species is shown in the third row of Figure 5.1. It can be seen that the magnetization curve shows a linear behavior for small applied fields. For larger fields saturation can be observed. The latter arises as more and more SPIOs reach a full alignment with the applied field. While this model yields qualitatively accurate results, it should be noted that measured magnetization curves often show some deviations from the Langevin model. These

are caused by the finite Brown- and Néel relaxation times. More accurate models can be developed using tools from non-equilibrium physics, such as the Langevin equation for systems far from equilibrium or the corresponding Fokker-Planck equation [104–106]. However, these are beyond the scope of this work.

5.3 Signal Encoding Principles

5.3.1 Particle Excitation

The signal encoding in MPI is based on the non-linear magnetization curve of SPIOs. Its basic mechanism is illustrated in the left column of Figure 5.1. To understand it, let us assume that the SPIOs are excited using a homogeneous sinusoidal field

$$\mathbf{H}_{\mathrm{E}}^{1d}(t) = H_{\mathrm{E}}^{x} \cos(2\pi f_{\mathrm{E}}^{x} t)\mathbf{e}_x, \tag{5.6}$$

with amplitude H_0^x and frequency f_{E}^x. As illustrated in the rows 2–4 of Figure 5.1, the particle magnetization follows the external field according to its magnetization curve modeled by Eq. (5.3). Importantly, the resulting magnetization dynamics is not sinusoidal due to the underlying non-linearity. As illustrated in the fourth row of Figure 5.1, the maxima and minima of the magnetization dynamics are flattened due to saturation in the underlying magnetization curve.

Next, let us consider the signal in a nearby receive-coil. As described in section 3.4, the signal induced by the particles is determined by the temporal derivative of the magnetization dynamics (see Eq. (3.16)). As a consequence, the induced particles signal shows peaks at the those times, where the magnetization experiences largest change, which is illustrated in the fifth row of Figure 5.1. However, it should be noted that the measured signal is superimposed by the signal induced by the excitation field $\mathbf{H}_{\mathrm{E}}^{1d}(t)$, which is orders of magnitude (10^6-10^9) larger than the particle signal. To separate the two signal contributions, the particle signal can be expanded in a Fourier series

$$u_l^P(t) = \sum_{k=-\infty}^{\infty} \tilde{u}_{l,k}^P \mathrm{e}^{2\pi \mathrm{i} k f_{\mathrm{E}} t}, \tag{5.7}$$

with

$$\tilde{u}_{l,k}^P = \frac{1}{T_R} \int_0^{T_R} u_l^P(t) \mathrm{e}^{-2\pi \mathrm{i} k f_{\mathrm{E}} t} \, \mathrm{d}t \tag{5.8}$$

and $T_R = 1/f_{\mathrm{E}}$. As illustrated in the sixth row of Figure 5.1 , this representation contains higher harmonics. In contrast, the harmonic nature of $\mathbf{H}_{\mathrm{E}}^{1d}(t)$ guarantees that the excitation field only contributes to the base frequency f_{E} of the measured signal. Thus, a signal attributable to the excited SPIOs can be measured by extracting the higher harmonics of the voltage induced in a receive coil.

The described excitation scheme is a powerful tool for the detection of SPIOs. A strength of it is that the particles can be detected with a high sensitivity [107] and that a direct quantification of the total magnetic moment is possible [17, 108]. Moreover, an analysis of the measured spectra can yield insights

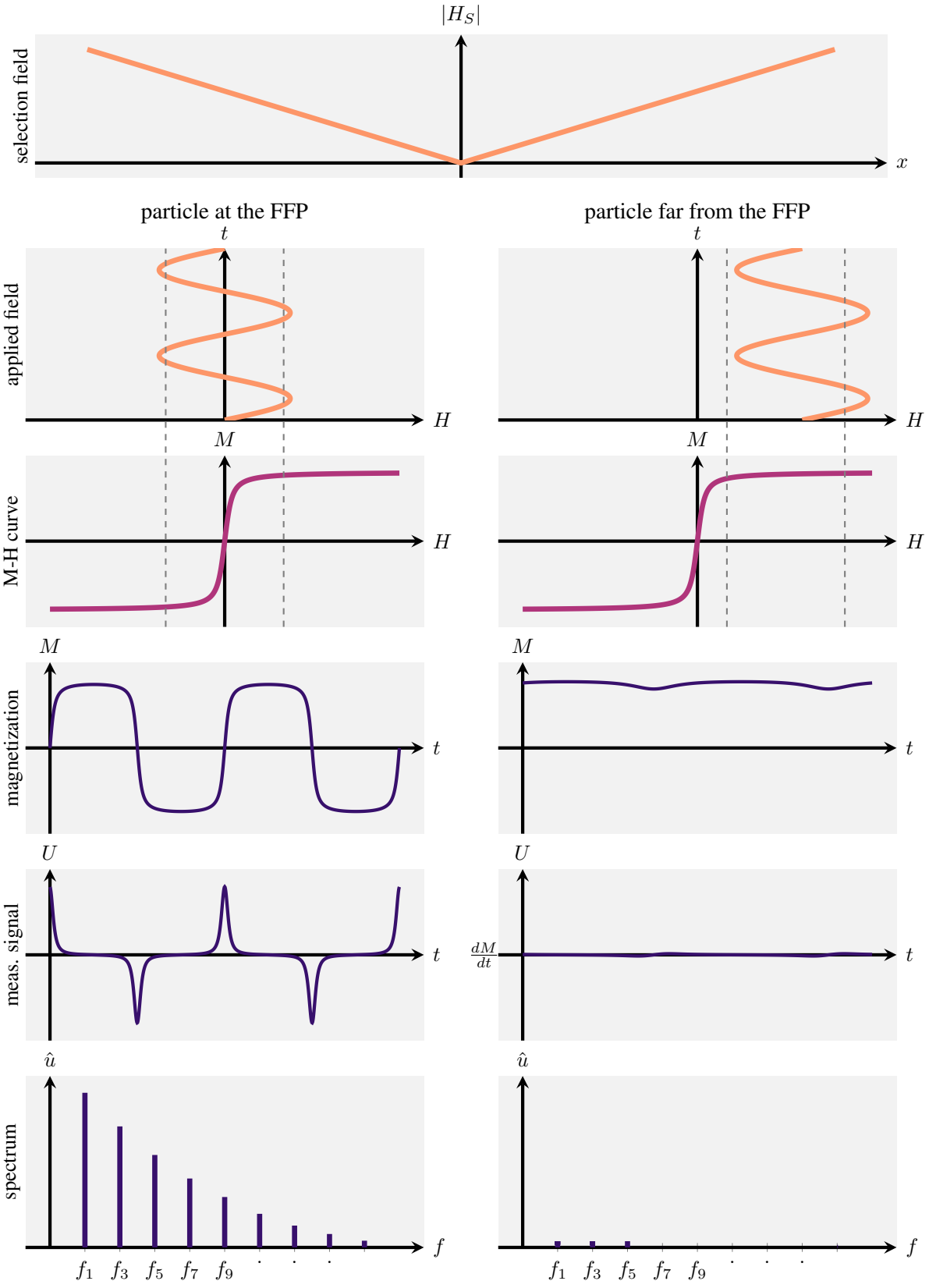


Figure 5.1: Illustration of the signal encoding in MPI. The SPIOs are excited by a superposition of a gradient field (shown in the first row) and a sinusoidal excitation field. This results in a periodically changing magnetization (third row). Due to the non-linear magnetization curve (fourth row), the induced voltage in a receive coil (fifth row) shows higher harmonics in the frequency domain (sixth row). Particles near the FFP exhibit a strong MPI signal (left column). Particles far from the FFP are saturated and exhibit only a weak MPI signal (right column).

not only on the particles but also on their environment. To be precise, we note that the form of the spectra is influenced by the particles Néel and Brown relaxation times as well as other characteristic properties such as particle anisotropy. Moreover, properties of the environment, such as temperature and viscosity, have an influence on the relaxation behavior of the particles and thus on the measured spectra [109–111].

5.3.2 Spatial Encoding

As discussed, the excitation scheme described in the previous section is a powerful tool for the detection and characterization of SPIOs. However, it does not contain a mechanism for encoding the location of the detected particles, which is a prerequisite for imaging applications. To include such a mechanism, MPI uses a time-independent selection field

$$\mathbf{H}_S(\mathbf{r}) = \begin{pmatrix} G_x x \\ G_y y \\ G_z z \end{pmatrix}, \tag{5.9}$$

which is superimposed to the excitation field $\mathbf{H}_{\mathrm{E}}^{1d}(t)$. Importantly the magnitude of $\mathbf{H}_S$ is zero at the origin and grows linearly with increasing distance from it, as is illustrated in Figure 5.1. This point, where the applied field vanishes is also called the field free point (FFP). As a consequence of the field configuration, SPIOs far from the origin are always in saturation, irrespective of the current excitation field. In turn, this implies a small particle signal, as is illustrated in the right column of Figure 5.1. Hence, this setup allows for a selective excitation where only particles in a small region with low field magnitude contribute significantly to the measured particle signal.

Using the selection field for spatial encoding, one is principally able to perform imaging. This can be achieved by using a small excitation field such that only particles in one voxel contribute to the measured signal. Following each excitation, the low field region must be shifted to allow imaging of the next voxel. The required shifts can be realized either via a mechanical shift or by applying a constant, homogeneous offset field. While this approach yields a simple mechanism for the imaging of SPIOs, it can lead to comparably long measurement times. The latter arise due to the need to translate the field configuration between particle excitation periods.

A faster approach to perform MPI has been proposed by Gleich and Weizenecker in the first paper on MPI [9]. It is based on the observation that the excitation field itself leads to a dynamic shift of the FFP. For instance, the combination of the excitation field (5.6) and selection field (5.9) leads to an oscillating FFP of the form

$$\mathbf{r}_{\mathrm{FFP}} = -\frac{H_{\mathrm{E}}^x}{G_x}\cos(2\pi f_{\mathrm{E}}^x t)\mathbf{e}_x. \tag{5.10}$$

As a consequence, fast MPI can be performed by applying a sufficiently strong drive field (DF), such that the FFP is translated across the entire field of view. From Eq. (5.10), one can also see that the resulting 1d-FFP-trajectory covers x-coordinates in the interval $\left[-\frac{H_0}{G_x}, \frac{H_0}{G_x}\right]$. This region covered by the FFP is also called drive field field of view (DF-FOV).

To generalize this approach to a 3d imaging setting, one can use a drive field of the form

$$\mathbf{H}_{\mathrm{E}}^{3\mathrm{d}}(t) = \sum_{i\in\{x,y,z\}} H_{\mathrm{E}}^{i} \cos(2\pi f_{\mathrm{E}}^{i} t)\mathbf{e}_i. \tag{5.11}$$

This also encompasses pure 2d imaging settings, which can be realized by setting one of the DF amplitudes to zero. To obtain a closed trajectory, the excitation frequencies are commonly chosen such that $\frac{f_{\mathrm{E}}^{y}}{f_{\mathrm{E}}^{x}}$ and $\frac{f_{\mathrm{E}}^{z}}{f_{\mathrm{E}}^{x}}$ are rational numbers. This leads to a 2d/3d Lissajous trajectory of the FFP, which is periodic in time and covers the DF-FOV. To furthermore achieve an isotropic resolution, many scanners use excitation frequencies of the form

$$\frac{f_{\mathrm{E}}^{y}}{f_{\mathrm{E}}^{x}} = \frac{N_{\mathrm{E}}}{N_{\mathrm{E}}+1} \tag{5.12}$$

for some $N_{\mathrm{E}} \in \mathbb{N}$. An example for a 2d Lissajous trajectory is shown in Figure 5.3.a). In direct extension to the 1d-case, the DF-FOV covered by a 2d trajectory along the x- and y-directions is a rectangle of size $\frac{2H_{\mathrm{E}}^{x}}{G_x} \times \frac{2H_{\mathrm{E}}^{y}}{G_y}$. Similarly, the DF-FOV for a 3d trajectory is a cube of size $\frac{2H_{\mathrm{E}}^{x}}{G_x} \times \frac{2H_{\mathrm{E}}^{y}}{G_y} \times \frac{2H_{\mathrm{E}}^{z}}{G_z}$.

From the results outlined previously, it follows that the FOV can be extended either by increasing the drive-field amplitude or by decreasing the gradient of the selection field. In practice, one needs to take into account that the DF amplitude can only be increased up to a certain point. Beyond that point patient safety can no longer be guaranteed due to peripheral nerve stimulation [112]. On the other hand, decreasing the selection field gradient has a negative impact on the achievable resolution. Thus, the drive field based spatial encoding incurs a trade-off between image resolution and FOV. Practical values for the field strengths depend on the application at hand. For instance, the Bruker preclinical MPI scanner (Bruker, Ettlingen) uses excitation frequencies of $(f_{\mathrm{E}}^{x}, f_{\mathrm{E}}^{y}, f_{\mathrm{E}}^{z}) = (2.5/102, 2.5/96, 2.5/99)$ MHz. Common imaging settings are a drive field amplitudes of up to $12\,\mathrm{mT}/\mu_0$ and selection field gradients of up to $(G_x, G_y, G_z) = (-1.25, -1.25, 2.5)$ T/(μ_0 m) [113]. When picking these values, the size of the corresponding drive field FOV is $(19.2 \times 19.2 \times 9.6)\,\mathrm{mm}^3$.

When determining suitable values for aforementioned imaging parameters, one should also keep in mind that the image resolution is not only determined by the size of the selection field gradients but also by the magnetization curve of the particles used in the imaging experiment. To better understand this dependency, we note that particles with a steep magnetization curve can be driven into saturation by smaller applied fields than particles with a more flat magnetization curve. Therefore, the ideal MPI tracer would consist of particles with a magnetization curve close to a step-function, as these allow for the highest resolution. In practice, the best behavior is obtained for SPIOs with an iron core size on the order of 25 nm and a correspondingly large mean magnetic moment $\|\bar{\boldsymbol{\mu}}\|$ [114]. In the Langevin model (5.3), it becomes clear that a large $\|\bar{\boldsymbol{\mu}}\|$ leads to a correspondingly steep magnetization curve and thus a high achievable resolution. Note however that experiments have shown no improvement in resolution when increasing the particle size beyond 25 nm. This is mainly caused by the particles finite relaxation times which limits their capability to follow the applied field [114].

5.4 MPI Signal Model

5.4.1 Induced MPI Signal

In order to reconstruct MPI images, the precise relationship between the measured signal and the underlying tracer distribution is of primary importance. To derive an expression for the measured signal, let us consider the voltage induced in a nearby receive coil with sensitivity profile $\mathbf{p}_l(\mathbf{r})$. Here the index l is used to index the different receive coils present in the experimental setup. The induced signal can readily be derived by combining Eq. (3.16) and Eq. (5.5), which leads to

$$u_l^P(t) = -\mu_0 \int_V c(\mathbf{r})\mathbf{p}_l(\mathbf{r}) \cdot \frac{\partial}{\partial t}\bar{\boldsymbol{\mu}}(\mathbf{H}(\mathbf{r},t))\,\mathrm{d}^3 r. \tag{5.13}$$

In practice, the measured signal is superimposed by the signal $u_E(t)$, which is induced by the excitation field itself. Due to the difference in magnitude of both signal contributions (10^6-10^9) and the limited bandwidth of modern ADCs, this makes a direct separation of the particle signal impossible. For this reason, the MPI receive chain comprises an additional band-stop filter, which is applied to the induced signal prior to digitization. Following the filtering, a low noise amplifier is used to bring the filtered signal into a range that can be processed by the ADC. Since filtering and amplification constitute a linear and shift invariant system, their combined action can be modeled as a convolution with a filter kernel $a(t)$. Thus, the filtered particle signal arriving at the ADC can be written as

$$u_l^F(t) = \int_0^T u_l^P(\tau)a(t-\tau)\,\mathrm{d}\tau.$$

An equivalent representation can be obtained in frequency space, where the convolution with $a(t)$ can be expressed as a multiplication with the Fourier coefficients

$$\tilde{a}_k = \frac{1}{T}\int_0^T a(t)\mathrm{e}^{-2\pi \mathrm{i} k t/T}\,\mathrm{d}t. \tag{5.14}$$

These coefficients are also referred to as transfer function. By combining (5.14) with our expression for the induced particle signal (5.13), we now end up with the following expression for the Fourier coefficients of the particle signal arriving at the ADC

$$\tilde{u}_{l,k}^F = -\hat{a}_k \frac{\mu_0}{T}\int_V c(\mathbf{r})\mathbf{p}_l(\mathbf{r}) \cdot \int_0^T \mathrm{e}^{-2\pi \mathrm{i} k t/T}\frac{\partial}{\partial t}\bar{\boldsymbol{\mu}}(\mathbf{H}(\mathbf{r},t),t)\,\mathrm{d}t\,\mathrm{d}^3 r. \tag{5.15}$$

Importantly, this relation confirms that the measured signal indeed depends linearly on the particle concentration. For completeness, we note that this model relies on the assumption of negligible particle-particle-interactions. In practice, this assumption is valid except for the case of very high particle concentrations.

5.4.2 The MPI System Matrix

The MPI signal equation (5.15) can be simplified further by separating the parameters related to the MPI system from the underlying tracer distribution. This leads the following representation of the MPI signal model.

Lemma 5.1. *The Fourier coefficients of the MPI signal induced by SPIOs with distribution $c(\mathbf{r})$ can be described by a linear model*

$$\tilde{u}_{l,k}^F = \int_V \tilde{s}_{l,k}(\mathbf{r}) c(\mathbf{r}) \, \mathrm{d}^3 r. \tag{5.16}$$

Here

$$\tilde{s}_{l,k}(\mathbf{r}) = -\hat{a}_k \frac{\mu_0}{T} \mathbf{p}_l(\mathbf{r}) \cdot \int_0^T \mathrm{e}^{-2\pi \mathrm{i} k t/T} \frac{\partial}{\partial t} \bar{\boldsymbol{\mu}}(\mathbf{H}(\mathbf{r}, t)) \, \mathrm{d}t. \tag{5.17}$$

denotes the MPI system function, which contains all parameters related to the MPI image acquisition.

In practice, MPI data is processed on a computer and images are reconstructed using a discrete grid. Therefore, we now assume the imaging grid (2.9) and perform discretization as described in section 2.4. Doing so yields the discrete forward model

$$\tilde{\mathbf{u}}_l = \mathbf{S}_l \mathbf{c}, \tag{5.18}$$

where we denote the voltage signal as $\tilde{\mathbf{u}}_l = (\tilde{u}_{l,k}^F)_{k=1,\dots,K}$ and the particle concentration as $\mathbf{c} = (c(\mathbf{r}_n))_{n=1,\dots,N}$. Moreover, we define the system matrix

$$\mathbf{S}_l = \left(\tilde{s}_{l,k}(\mathbf{r_n}) \Delta V\right)_{k=1,\dots,K; n=1,\dots,N}. \tag{5.19}$$

When a measurement is performed using multiple receive coils, an analogous matrix-vector form can be obtained, by defining the concatenated voltage vector $\tilde{\mathbf{u}} = \left(\tilde{\mathbf{u}}_1^\mathsf{T} \cdots \tilde{\mathbf{u}}_L^\mathsf{T}\right)^\mathsf{T}$ and the corresponding system matrix $\mathbf{S} = \left(\mathbf{S}_1^\mathsf{T} \cdots \mathbf{S}_L^\mathsf{T}\right)^\mathsf{T}$.

Structure of the MPI System Matrix

To investigate the structure of the MPI system matrix, we first consider the special case of a 1d-excitation and a homogeneous coil sensitivity $\mathbf{p} = p\mathbf{e}_x$. In this case, the Fourier transform in (5.17), can be computed analytically as was shown in [115]. The result is summarized in the following Lemma.

Lemma 5.2. *For a 1d-excitation of the form (5.6) and ideal fields, the MPI system function can be written as*

$$\tilde{s}_{1,k}(x) = -\frac{2\mu_0 p a_k \mathrm{i}}{T} (\bar{\mu}'(G_x \cdot) * \upsilon_{k-1}(G/H_E^x \cdot))(x). \tag{5.20}$$

where $\bar{\mu}'(\cdot) = \frac{\partial \|\bar{\boldsymbol{\mu}}\|}{\partial H}$ *and* $\upsilon_k(\cdot) = \sin((k+1)\arccos(\cdot))$. *If one further assumes an ideal, step-like magnetization curve, this expression further simplifies into*

$$\tilde{s}_{1,k}(x) = -\frac{2\mu_0 p \mu a_k \mathrm{i}}{T} \upsilon_{k-1}(G/H_E^x x), \tag{5.21}$$

where μ *denotes the step size of the magnetization curve.*

As a side remark, we note that $\upsilon_k(\cdot)$ can also be written as $\upsilon_k(x) = U_k(x)\sqrt{1-x^2}$, where U_k denotes the Chebyshev polynomial of the second kind and degree k. Thus, for 1d-excitations, the structure of the system matrix shows an oscillatory structure which can be well described using Chebyshev polynomials of the second kind.

A close relation to aforementioned Chebyshev polynomials can also be observed for higher-dimensional Lissajous-type excitations. To explore this relationship, we now consider a 2d Lissajous excitation along the x- and y-direction as given in Eq. (5.11). In this case, the non-linear particle magnetization leads to a particle signal which contains harmonics of both excitation frequencies f_{E}^x and f_{E}^y. As described in Ref. [100], this frequency mixing can be described using intermodulation theory [116, 117]. Based on the latter, the signal at a frequency f_k can be described in terms of the harmonic frequencies $m_x f_{\mathrm{E}}^x$ and $m_y f_{\mathrm{E}}^y$, which are linked to f_k via the relationship

$$f_k = m_x f_{\mathrm{E}}^x + m_y f_{\mathrm{E}}^y. \tag{5.22}$$

Here the $m_x \in \mathbb{Z}$ and $m_y \in \mathbb{Z}$ are known as mixing factors. If we further assume excitation frequencies of the form (5.12), the explicit relationship between the linear frequency index and the mixing factors is given by [100]

$$k(m_x, m_y) = (N_{\mathrm{E}} + 1)m_x + N_{\mathrm{E}} m_y. \tag{5.23}$$

With these mixing factors at hand, it turns out that a good approximation to the frequency components detected on the x- and y-channel is given by [118]

$$\tilde{s}_{1,k}(x,y) \approx \alpha_{1,k}\Big(\kappa(x,y) * \upsilon_{m_x-1}(G_x/H_{\mathrm{E}}^x x)\upsilon_{m_y}(G_y/H_{\mathrm{E}}^y y)\Big) \tag{5.24a}$$

$$\tilde{s}_{2,k}(x,y) \approx \alpha_{2,k}\Big(\kappa(x,y) * \upsilon_{m_x}(G_x/H_{\mathrm{E}}^x x)\upsilon_{m_y-1}(G_y/H_{\mathrm{E}}^y y)\Big), \tag{5.24b}$$

with $\alpha_{1,k} \in \mathbb{C}$ and $\alpha_{2,k} \in \mathbb{C}$ denoting appropriate proportionality constants and κ being a kernel that depends on the magnetization curve of the particles at hand. As a result, the rows of the MPI system matrix can indeed be approximated by tensor products of Chebyshev polynomials, given that the kernel κ is sufficiently similar to the δ-distribution. Analogously to the 1d case, this approximation becomes better the steeper the magnetization curve.

Naturally, this product-like structure can also be found in the MPI system matrix. To illustrate this, Figure 5.2 shows various frequency components of a 2d system matrix arranged according to their mixing factors. Below each pattern, the corresponding tensor-product of Chebyshev polynomials is shown. Note

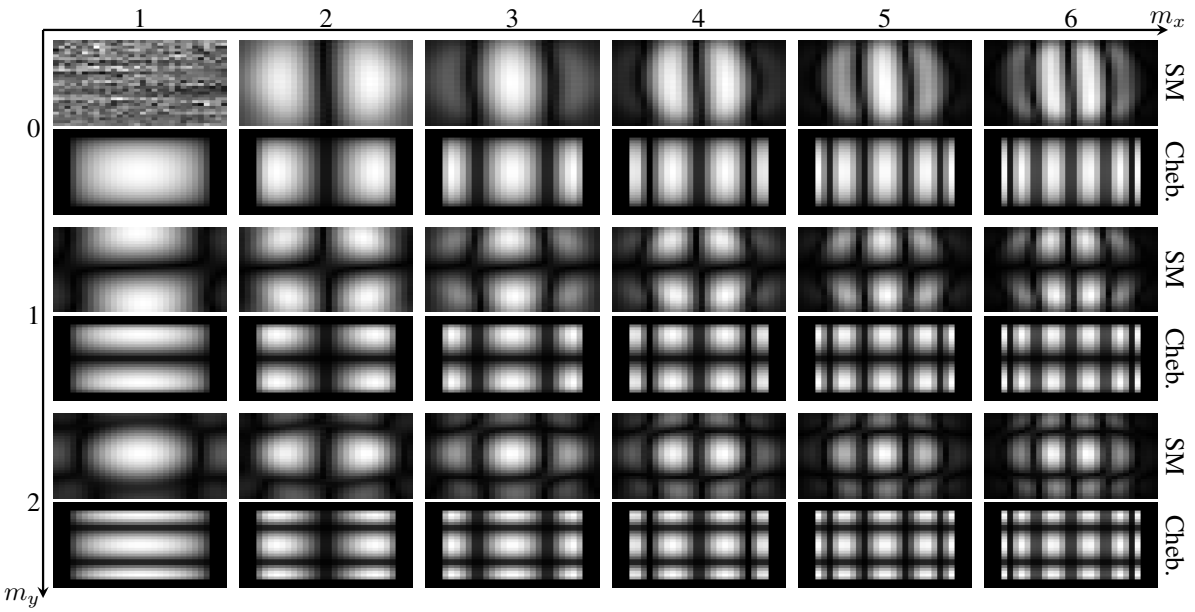


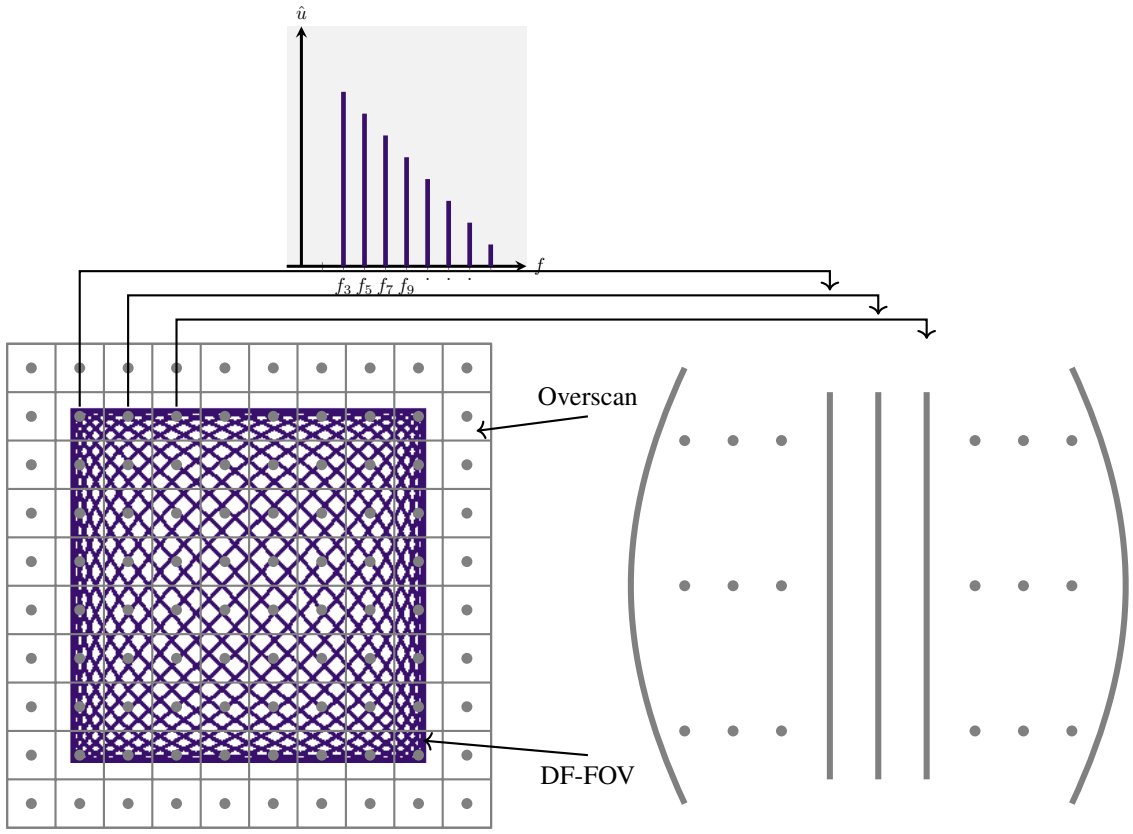
Figure 5.2: Illustration of frequency components with different mixing factors (m_x, m_y) for the x-channel of a measured 2d system matrix. The bottom part of each panel shows the corresponding approximation in terms of Chebyshev-polynomials.

that Figure 5.2 uses a representation where each row of the system matrix is interpreted as an $(N_x \times N_y)$-matrix. This representation is beneficial because it takes all spatial dimensions into account in the same way as the underlying system function. In an analogous manner, the rows of a 3d system matrix can be interpreted as 3-way tensors of the appropriate size.

Determination of MPI System Matrices

Based on the previous discussion, the system matrix $\mathbf{S}$ provides us with a suitable model to perform image reconstruction for settings with negligible inter-particle interactions. In practice, it should be noted that deviations from the Langevin model can often be observed, e.g. due to finite relaxation times. Moreover, real MPI systems display certain imperfections, such as inhomogeneity of the drive and selection fields, which lead to deviations from the ideal signal model. For this reason, a calibration scan is typically used to determine the system matrix experimentally. In this calibration scan, a small sample of tracer material (called delta sample) is subsequently placed in the center of each voxel of the imaging grid and the resulting particle signal is measured, as is illustrated in Figure 5.3. This procedure has the advantage that it is very generic and it provides a stable way to describe the response of the imaging system under a variety of different conditions. For instance, dedicated system matrices can be obtained for different tracers, different scanner parameters or under different physical conditions.

Mathematically, the described calibration scheme corresponds to a set of measurements using the canonical measurement basis (2.14). When comparing this with the linear measurement model (5.18),



a) Cell-centered grid b) System matrix

Figure 5.3: Illustration of the calibration process in MPI. A delta sample is placed at the center of each point of the imaging grid (a) and the resulting spectra are measured. The system matrix (b) is obtained by placing each spectrum in to the column corresponding to the respective point on the imaging grid.

one finds that each of the resulting spectra corresponds to one column of the underlying system matrix. When considering the k^{th} frequency component $\mathbf{s}_{k,:}$, this leads to the measurement model

$$\mathbf{y}_k = \mathbf{H}_\Omega \mathbf{s}_{k,:} + \boldsymbol{\eta}_k. \tag{5.25}$$

Here $\mathbf{H}_\Omega$ is a sampling matrix of the form (2.14) and $\mathbf{y}_k$ denotes the measured voltages at the given frequency normalized by the tracer concentration. Finally $\boldsymbol{\eta}_k$ denotes the measurement noise, which we assume to be normally distributed.

System Matrix Recovery

A drawback of the calibration-based measurement of MPI system matrices are the long measurements required. Using a preclinical Bruker 15/20FF MPI scanner(Bruker Corp., Ettlingen, Germany), 2d system

matrices can be acquired in about half an our. However, the measurement of a 3d system matrix can easily take more than a day [113]. This is particularly problematic when separate system matrices need to be acquired, such as when performing multi-patch imaging. These long calibration times are particularly problematic because it is challenging to keep MPI scanners stable over such long periods of times. On the current generation of scanners, instabilities can lead to corrupted measurements due to imperfections in the electronics or software components of the scanner at hand. For some MPI scanners, such as the human-sized low-field brain imager presented in [97], coil heating becomes an issue when measurements are performed over long time periods. In this case the required cooling times render the measurement of large FOV 3d system matrices infeasible.

To address these challenges, to reduce the number of calibration measurements and apply CS-based approaches to recover the underlying system matrix rows [28, 29, 119]. Based on the discussion in section 2.7, the application of CS requires knowledge of a transform $\mathbf{\Psi}$ which renders the system matrix rows sparse and a sampling pattern which is incoherent with respect to the sparsity basis. For the MPI system matrix rows, sparsifying transforms can be derived due to their characteristic oscillatory structure, which motivates a sparsity basis based on periodic functions. In fact, all past works on the application of CS for the recovery of system matrices have set $\mathbf{\Psi}$ to be either the DCT or the DFT. To obtain a robust method, this was combined with a pseudo-random sampling pattern, which can be motivated by the recovery guarantees summarized in subsection 2.7.4. Putting everything together, recovery of a given system matrix row can then be achieved by solving the inverse problem

$$\underset{\mathbf{s}_{k,:}}{\operatorname{argmin}} \left\|\mathbf{\Psi}\mathbf{s}_{k,:}\right\|_1 \text{ subj. to } \frac{1}{2}\left\|\mathbf{H}_\Omega\mathbf{s}_{k,:} - \mathbf{y}_k\right\|_2^2 < \varepsilon^2, \tag{5.26}$$

where ε^2 is an estimate for the measurement noise.

5.5 Image Reconstruction

Having described how MPI encodes information about the SPIO distribution $c(\mathbf{r})$ at hand, this section discusses how the latter can be reconstructed from a given set of MPI measurements. A generic approach is to assume a linear model, such as the system-matrix-based description in section 5.4. This allows to formulate image reconstruction using the statistical framework described in chapter 2. In this case, image reconstruction boils down to an inverse problem, which can be solved using optimization methods. A second approach is the so-called x-space approach, which operates in time space and directly uses the location of the FFP to attribute tracer concentration to the underlying voxels. In this work we solely focus on the system-matrix-based approach. For more details on x-space image reconstruction, we refer the reader to Ref. [120].

5.5.1 System-Matrix-Based Image Reconstruction

Having established the linear relationship between particle concentration and the measured signal, one can now use the techniques introduced in section 2.5 to estimate the underlying tracer distribution $\mathbf{c}$ from a given a MPI measurement $\tilde{\mathbf{u}}$. The most popularly used image reconstruction techniques in MPI are implicitly based on the assumption of Gaussian measurement noise. Thus, we assume the measured voltage signal to be distributed as $\tilde{\mathbf{u}} \sim \mathcal{N}(\mathbf{S}\mathbf{c}, \sigma_u^2 \mathbf{1})$. In practice, one should be aware that measurement noise has multiple origins in MPI. Examples include thermal noise induced by the electric resistance in the receive chain, noise induced by the used amplifier and noise induced by the quantization in the ADC. Furthermore, noise can be generated by eddy currents which are induced in the tissue of the patient. In such a setting, the assumption of Gaussian measurement noise can be justified by the central limit theorem, as discussed in section 2.4.

In addition to the measurement noise, we also assume a Gaussian prior distribution with diagonal covariance matrix for the concentration vector $\mathbf{c} \sim \mathcal{N}(\mathbf{0}, \sigma_c^2 \mathbf{1})$. With this at hand, image reconstruction can be performed by computing the corresponding MAP estimator, i.e. by solving the Tikhonov regularized least squares problem

$$\underset{c}{\operatorname{argmin}} \|\mathbf{S}\mathbf{c} - \tilde{\mathbf{u}}\|_2^2 + \lambda \|c\|_2^2 , \tag{5.27}$$

with the regularization parameter $\lambda = \frac{\sigma_u^{2K}}{\sigma_c^{2N}}$.

As outlined in chapter 2, a solution to problem (5.27) can be directly computed. An alternative approach is the use of iterative methods. In particular, the Kaczmarz method [121, 122] is a very popular tool for MPI reconstruction. As shown in [121], this methods often converges within the order of 10 iterations. Due to its fast convergence and the simplicity of the method, the Kaczmarz algorithm can easily achieve the speeds required for a real-time reconstruction of an image series. This is the reason why the Kaczmarz method is the most popular method for MPI image reconstruction.

For completeness, we note that more advanced reconstruction schemes have also been proposed for MPI [123–125]. Most of these methods replace the ℓ_2-regularization term in (5.27) by other regularization terms such as total variation regularization, or ℓ_1-regularization in the Wavelet domain. The idea underlying these methods is to use regularization functions that more closely match the statistics of the underlying images. For instance, total variation regularization promotes solutions that have sparse image gradients. Their use is motivated by the observation, that natural images typically have a rather small number of edges. In contrast, Wavelet-ℓ_1-regularization exploits the observation that the wavelet transform of natural images usually displays only a small number of non-negligible coefficients. As demonstrated in [123–125], these more advanced regularization techniques can indeed provide reconstructions with a reduced amount of image blurring and a reduced bias in the obtained concentration maps. On the other hand, these methods typically involve the minimization of a non-smooth optimization functional. This makes it harder to achieve real-time reconstructions.

5.6 Challenges

Using the principles outlined in this chapter, MPI can be used to measure the distribution of SPIOs with high spatial resolution, high temporal resolution as well as a high sensitivity. Nevertheless, the complex underlying physics as well as security requirements lead to challenges, some of which will be addressed in this work.

As discussed in subsection 5.4.2, one challenge of MPI is the long time that is needed for the measurement-based determination of system matrices. To alleviate these issues, one can use CS to recover system matrices from a set of undersampled measurements. While this approach has been shown to work, previous works have only focused on simple sparsifying transformations such as the DCT. Similarly, all previous works have focused on random sampling patterns with a uniform sampling density. Hence, further improvements can be expected both by exploring other forms of regularization and by developing methods to adapt the sampling patterns to the expected structure of a given system matrix. These challenges will be addressed in chapter 8 and chapter 10.

Another challenge with MPI is to achieve imaging with both a large FOV and a high resolution. As discussed in subsection 5.3.2, the imaging resolution is mainly determined by the gradient of the selection field, whereas the FOV is determined by the ratio of the drive field amplitude and the selection field gradient. To enlarge the field of view beyond the limits imposed by the selection field and drive field, one can use sequential measurements of multiple patches. After each measurement, a static offset field is applied in order to shift the DF-FOV to another position. While this allows to enlarge imaging volume, temporal resolution is reduced due to the sequential measurement of all patches. A further issue is that dedicated system matrices must be acquired for each image patch. Hence, the system calibration becomes more time consuming.

Based on these challenges, a further interesting research direction concerns the recovery of system matrices in a multi-patch setting. This setting bears additional challenges, because the structure of the system matrix rows is often distorted by field imperfections. On the other hand, this setting also offers the potential to achieve even larger undersampling factors by exploiting the strong inter-patch-correlations in the structure of the system matrices.

Finally, a lot of research is directed towards the field of multi-contrast MPI [13, 126]. The latter allows the simultaneous imaging of multiple typos of tracers during one imaging experiment. This field of research has a high impact because it opens up the door for a large number of applications, such as the tracking of surgical instruments [14]. Interestingly, multi-contrast MPI can also be used to probe the environmental conditions of the MPI tracer, thus enabling applications such as viscosity mapping [127] and temperature mapping [128].

Part II

Compressed Sensing

6 Image Reconstruction Frameworks

This chapter discusses the development of image reconstruction frameworks, which are both performant and easily extendable. Such frameworks are an important tool for the development and the practical application of new reconstruction algorithms. To facilitate their development, we present a generic optimization framework, which is implemented in the programming language Julia and provides the basic building blocks to formulate and solve image reconstruction problems. Based on this functionality, we further discuss two image reconstruction frameworks, for the modalities MRI and MPI, of which I was a co-developer. This work was published in [J3] and [J4].

6.1 Motivation

In recent times, many imaging modalities, including MRI and MPI, have profited from the introduction of advanced, iterative image reconstruction methods. Particularly in MRI, techniques such as CS and PI have not only improved acquisition times and spatial resolution, but also stimulated the development of new techniques for the quantitative imaging of tissue parameters, such as relaxation rates, magnetic susceptibility or apparent diffusion coefficients [8, 56, 129]. As discussed in section 4.5, the use of sophisticated algorithms can thus mitigate limitations in the acquisition hardware, albeit at the expense of a more computationally intensive reconstruction pipeline. For these reasons, the availability of efficient reconstruction libraries has become a key requirement for the development of new reconstruction methods and their deployment in actual applications.

When developing reconstruction algorithms, researchers usually face two conflicting challenges. On one hand, computation times should be as small as possible. On the other hand, the programming environment should allow for an easy translation of mathematical formulations into programming code. The first of these goals is often addressed by working in a low-level programming environment. While this allows the generation of efficient machine code, it also leads to a high complexity of the software. Conversely, the second goal, can be addressed by working in a high-level programming environment, such as Matlab or Python/Numpy [37], albeit at the cost of computational performance.

A popular approach to reconcile both goals is to use a hybrid approach, where only hot loops are implemented in a low-level language, while the remaining part of the code is implemented in a high-level language. Commonly used examples for this approach are Python with its different ways to call

C code (e.g. ctypes), as well as Matlab, which allows to incorporate C/C++ using its Mex interface. On a higher level, one finds that entire frameworks can be built using the hybrid approach. Popular examples include Numpy [37], which is mainly written in C, and machine learning frameworks such as PyTorch [38], which provide Python bindings to facilitate the use of the underlying C/C++ routines.

While the hybrid approach has many advantages it also has certain limitations, which the authors in [41] refer to as the two-language problem. For instance, the low-level implementation is capsuled away from the high-level user interface, which complicates the implementation of custom algorithms. Secondly, bridging from one language to another often incurs high computational costs and only amortizes for large problem sizes. As a result, code is often vectorized to minimize the number of calls into the low-level language. While vectorization is not a bad thing per se, there are a number of examples, such as the fast Fourier transform (FFT), which lend themselves more to loop-based implementations than to vectorization.

Based on the previous discussion, there is a need in algorithmic research for reconstruction frameworks yielding high performance while allowing for easy prototyping of new algorithms. At the same time, such frameworks should be implemented in a single language to avoid the two-language problem. Motivated by these issues, this work describes a flexible and extendable mathematical optimization framework that has been developed in the programming language Julia and aims to address aforementioned aspects. The developed framework provides the basis for the MRI reconstruction framework, which was developed by Tobias Knopp and myself [J3] and whose central part is the Julia package *MRIReco.jl*. Similarly, the described framework is also used in the MPI reconstruction package *MPIReco.jl*, which was developed by Tobias Knopp in cooperation with Martin Möddel, Patryk Szwargulski, Florian Griese, Marija Boberg and myself. A strength of the developed framework is its flexibility and extensibility, which makes it a valuable tool for the development of image reconstruction algorithms even across imaging modalities.

To put this work into further perspective, we note that most popular MRI reconstruction frameworks follow the hybrid approach. Examples include Gadgetron [39] and BART [40], which both have many features and have cores written in C/C++ with bindings to different high-level languages. An exception is SigPy [130], which is a high-level package entirely written in Python and allows for rapid prototyping of MRI reconstruction algorithms. For the field of MPI, *MPIReco.jl* was the first published open-source reconstruction framework. A recently developed framework is MPIRF, which is implemented in Python and makes use of the hybrid framework Numpy [131]. A common point of most other image reconstruction frameworks (with the exception of SigPy) is that they only provide a selection of image reconstruction methods with a fixed set of options and limited customizability. This often complicates the implementation of custom algorithms, which is one of the objectives of this work.

6.2 Software Design Considerations

6.2.1 Design Goals

The development of image reconstruction frameworks is usually guided by a set of design goals. These goals have an influence not only on the programming language to be used for the implementation but also on the underlying software infrastructure used to solve the given image reconstruction problems. The frameworks *MRIReco.jl* and *MPIReco.jl* both aim to fulfill the following set of goals, which were formulated and further explained by the developers of Gadgetron [39]

1. Free / open
2. Modular
3. Flexible
4. Cross platform
5. High performance
6. Facilitate prototyping
7. Facilitate deployment

Currently, the deployment aspect is only addressed in *MPIReco.jl*. However, both frameworks have the following additional design goals:

8. *Reuse of components:* We aim to use existing software components whenever they are suitable to the task and appropriately maintained. This also implies putting functionality that is not specific to the given modality into dedicated packages, in order to allow their reuse in other applications.
9. *Hackability:* We aim for a small gap between being a software user and a software developer. Thus, it should be easy to access low-level parts of the software, make changes to it and implement extensions of the code.
10. *Accessibility:* It should be easy to install, access and modify the software code.
11. *Testing:* The code should be properly tested using continuous integration services.

We note that the design goals 10 and 11 are also followed by most other image reconstruction frameworks. However, two-language solutions often encounter limits as to how well points 8 and 9 can be fulfilled. For instance, hackability is hard to achieve when both the high-level and the low-level languages need to be known in order to modify a given framework.

As is illustrated by the different imaging modalities considered in this thesis, even frameworks from different fields are based on similar techniques and share a large number of generic buildings blocks. To enable exchange between these fields and foster synergy effects, the reuse of components is a central aspect. By providing building blocks, such as linear algebra tools or optimization methods, in dedicated packages these can be reused by many different applications. Additionally, developers are enabled to reuse building blocks implemented by experts of the given field, thus maximizing performance while minimizing potential errors.

6.2.2 Julia

Guided by the previously outlined design goals, the developed frameworks are implemented using the Julia programming language / environment [41]. The latter was invented with the aim of solving the two-language problem, which makes it a suitable tool for our purposes.

To achieve performance, Julia uses a just-in-time compiler, based on the LLVM compiler infrastructure [132]. This is combined with a sophisticated type system, which allows the compiler to identify type stable code fragments thus enabling the generation of efficient machine code.

To achieve ease-of-use, the syntax of Julia is inspired by the well known Matlab syntax, which guarantees a high degree of familiarity for many researchers. At the same time it should be noted that Julia can be both high-level and low-level. This allows to keep hot code paths type-stable in order to maximize performance, whereas a more convenient, dynamic programming style can be used in other parts. Additionally, we note that C/C++ code can be called with no overhead, which simplifies the integration of other C/C++-based programming code.

Another distinct feature of Julia is its emphasis of the concept multiple dispatch. According to the latter, a function can have methods providing implementations for different types of any of the positional arguments. When calling the function, Julia automatically chooses the most specific method applicable to the given set of arguments. This is in contrast to many other climbing languages, where dispatch happens with regard to only one function argument Multiple dispatch proves to be a powerful paradigm for the structuring of code. In the view of image reconstruction frameworks, it makes it easy to implement a large number of high- and low-level methods, while retaining an interface that is simple and can easily be used and extended.

Finally, Julia features a powerful, built-in package manager to handle all the dependencies of a Julia-package. As a consequence, a large number of dedicated packages can easily be managed, which allows for a fine-grained modularization across packages. This modularization is pushed by a large community of package developers, which enrich the power of the Julia programming environment substantially. This aspect of Julia is particularly important to achieve an efficient reuse of components (design goal 8).

6.2.3 Image Reconstruction Components

When taking a high-level point of view, most image reconstruction frameworks can be modeled using a set of generic building blocks, which are illustrated in Figure 6.1 along with the associated data flow. For the reconstruction of imaging data, the first requirement is the ability to read measurement data from the files they are stored in. This task is specific to the imaging modality at hand and usually involves implementing readers for the file formats used by the imaging devices. Having obtained access to the raw measurement data, one may face a second problem because the raw data is often written to the file in the order of acquisition. Commonly this form is not suitable for imaging reconstruction. Therefore, a set of preprocessing steps might be required to convert the raw acquisition into the processed form to be

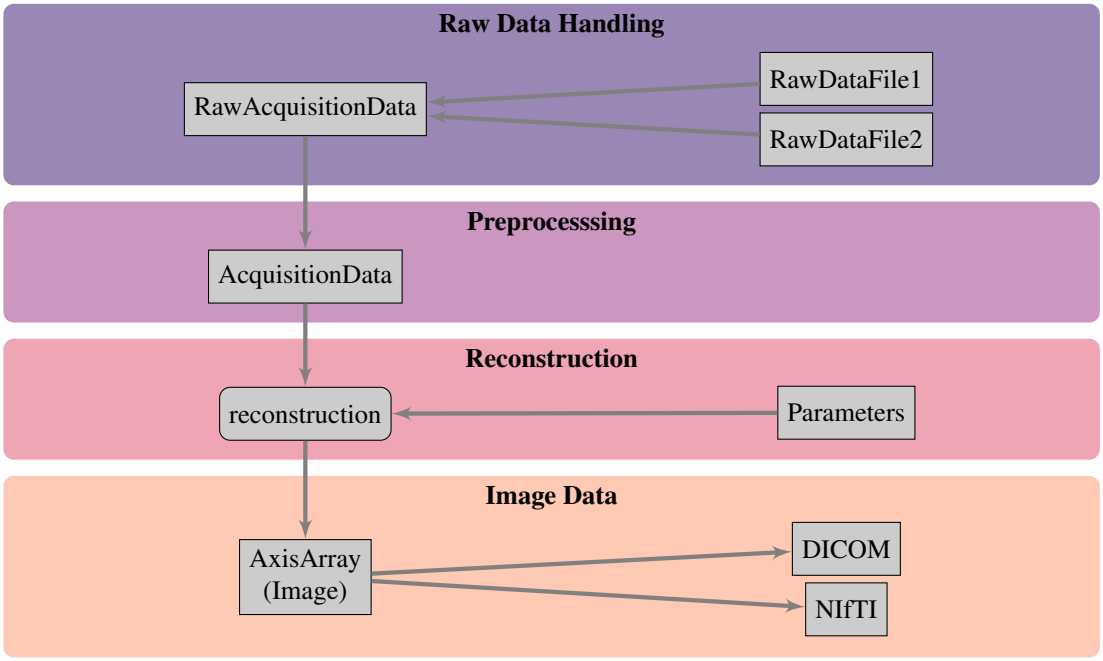


Figure 6.1: Overview of the typical components and data flow in image reconstruction frameworks.

used for image reconstruction. Similar to the data loading, these steps are specific to the given imaging modality.

Following the preprocessing, image reconstruction can be performed to recover the image associated with the measured data. In contrast to the previous steps, image reconstruction often follows a more generic scheme. As will be discussed in the following section, this results from the fact that image reconstruction is commonly formulated as an inverse problem. Hence, image reconstruction can be performed using generic mathematical optimization methods. In this case, only the form of the imaging operator and the exact formulation of the reconstruction problem are specific to the imaging modality at hand.

The reconstructed images are usually saved to disk for future reference. In addition to that, many applications require a further post-processing in order to extract the desired information. In Julia, images can be exported into popular image formats, such as DICOM or NIfTI by making use of the corresponding packages *DICOM.jl* and *NIfTI.jl*. Moreover, a great variety of image processing methods are available via packages provided by the JuliaImages organization.

Making use of the described structure, Julia with its package manager offers the opportunity to develop lightweight image reconstruction frameworks. This can be achieved by only focusing on the parts of the pipeline that are specific to the imaging modality at hand. As described before, this reduces the risk of errors which might arise when reimplementing popularly used standard methods. A second advantage is that the developed frameworks lose some of their complexity due to the high modularity. This facilitates the work of algorithmic researchers because it allows them to focus on the actual development

of algorithms without getting distracted by issues such as the technicalities of some modality-specific preprocessing steps.

6.3 Optimization Framework

A central aspect of this work is the development of the building blocks, which commonly appear in the actual reconstruction layer of different image reconstruction frameworks. These can be used by algorithmic researchers to compose custom reconstruction algorithms. Whereas, the description of measured data is specific to the given imaging modality, image reconstruction is usually performed by solving an inverse problem as has been outlined in chapter 2. In this work we mainly focus on regularized least squares problems of the form

$$\underset{\mathbf{m}}{\operatorname{argmin}} \ \|\mathbf{u} - \mathbf{S}\mathbf{m}\|_2^2 + \mathcal{R}(\mathbf{m}), \tag{6.1}$$

where the regularization function $\mathcal{R}$ expresses prior knowledge about the solution. This class of problems is sufficiently large to capture state-of-the-art image reconstruction methods from many imaging modalities including MRI and MPI.

The most common solvers for these types of problems alternate between two types of steps in order to approximate the solution. In one step, one searches for a solution candidate which reduces the data fidelity cost $\|\mathbf{u} - \mathbf{S}\mathbf{m}\|_2^2$. In a separate step, one aims to reduce the value of the regularization term $\mathcal{R}(\mathbf{m})$. Solvers of this type can be modeled in terms of three basic building blocks, each of which will be discussed in more detail in the following

1. *Linear Operators:* The linear operator $\mathbf{S}$ is required to minimize the data fidelity cost. If $\mathcal{R}$ also contains a linear transformation, the latter is needed to compute the update reducing the regularization cost.
2. *Proximal Maps:* The proximal map is used to reduce the cost associated with $\mathcal{R}$. It can be defined as [133]
$$\operatorname{prox}_{\mathcal{R}}(\mathbf{m}) := \underset{\mathbf{u}}{\operatorname{argmin}} \frac{1}{2} \|\mathbf{u} - \mathbf{m}\|_2^2 + \mathcal{R}(\mathbf{u}).$$
3. *Solvers:* The actual optimization algorithm used to solve (6.1) based on aforementioned linear operators and proximal maps.

To make these building blocks available for various applications, we developed the Julia packages *RegularizedLeastSquares.jl*[1] and *SparsityOperators.jl*[2] both of which are hosted on GitHub. *RegularizedLeastSquares.jl* provides popular solvers and regularization functions, whereas *SparsityOperators.jl* contains a collection of linear operators, which are commonly needed in different imaging modalities. To illustrate their use in the current Julia ecosystem, Figure 6.2 shows the dependencies of the developed optimization and image reconstruction frameworks.

[1] https://github.com/tknopp/RegularizedLeastSquares.jl
[2] https://github.com/tknopp/SparsityOperators.jl

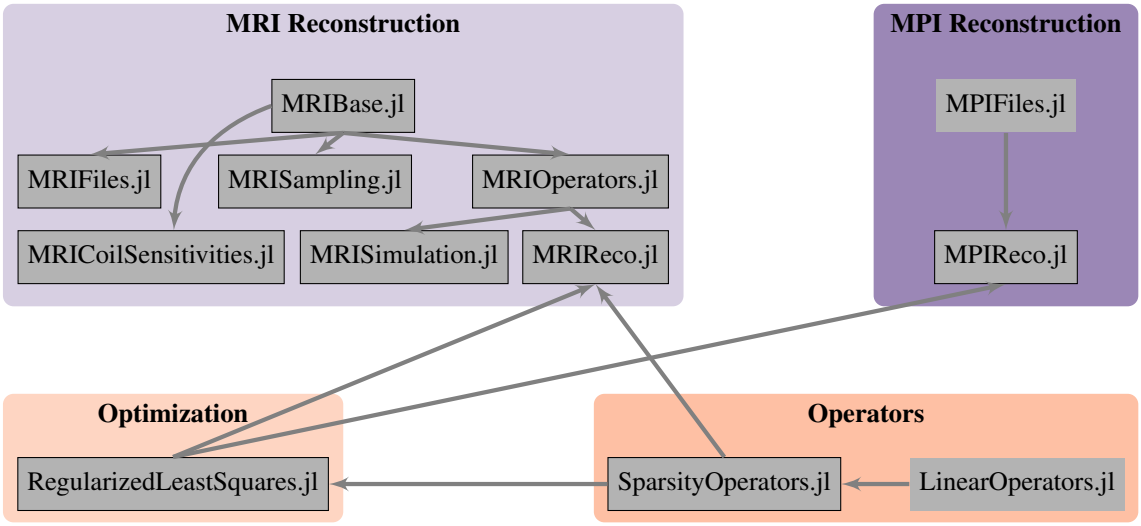


Figure 6.2: Overview of the dependencies of the developed image reconstruction frameworks. The orange blocks correspond to the mathematical optimization framework underlying the modality-specific image reconstruction frameworks. Packages that were developed with my collaboration are marked by black rectangles

Linear Operators

As described above, linear operators can arise in different places of an image reconstruction algorithm. The most central operator is the imaging operator $\mathbf{S}$, which is specific to the given imaging modality. For this reason it is usually implemented in the specific reconstruction framework for each modality. Nevertheless, it can often be composed of multiple generic building blocks. As an example, the imaging operator for a Cartesian MRI sequence can be formulated solely in terms of diagonal matrices, FFTs and a sampling operator, as is evident from Eq. (4.27). Secondly, linear operators often arise as part of the regularization term $\mathcal{R}$. As an example, consider the CS reconstruction problem (4.34), where $\mathbf{\Psi}$ transforms the image into a domain where the image is sparse.

To facilitate their application, *SparsityOperators.jl* contains implementations of various linear operators that commonly arise in image reconstruction. These include an `FFTOp` to perform FFTs and a `WaveletOp` to compute discrete Wavelet transforms. Moreover, a `WeightingOp` is provided for tasks such as sampling density compensation and a `SamplingOp` allows to model the subsampling commonly performed in CS.

An important aspect about the described operators is that all of them are implemented without forming their actual matrix representation. Often times, this is beneficial because it allows to improve upon the $\mathcal{O}(MN)$ complexity, which is required for computing the matrix-vector product with a matrix of size $M \times N$. For instance, the FFT has a computational complexity of $\mathcal{O}(N \log(N))$, which is a serious improvement given that the corresponding DFT-matrix is of size $N \times N$. Even more efficient implemen-

tations exist for diagonal matrices, sampling matrices and the discrete Wavelet transform, all of which can be evaluated in $\mathcal{O}(N)$.

Besides the matrix-free implementation, linear operators can be used just like matrices. For instance, a linear operator can be applied using the $*$-operators and its adjoint can be formed using the postfix $'$. Similarly, addition of operators can be performed using the operator $+$ and composition can be performed using the operator $*$. This allows for a simple translation of mathematical notation into code.

Proximal Maps

Infrastructure for regularization functions is capsuled into the package *RegularizedLeastSquares.jl*. Notably, this package contains the type `Regularization`, which stores all relevant information to apply a given regularization function in a solver. The central part of this type is a function to compute the proximal map associated with a given regularization function.

For convenience, *RegularizedLeastSquares.jl* implements several common regularization functions such as total variation, ℓ_1, ℓ_2 and low-rank regularization. The regularization function to be used can be easily generated by passing its name to the high-level constructor provided by the package. Moreover, it is easy to build custom `Regularization`. In this case, the main work to be done is the implementation of the associated proximal map.

Concerning the associated computational complexity, we note that separable functions, such as the ℓ_1 and ℓ_2 norm, have simple proximal maps which can be computed in an element-wise manner and with $\mathcal{O}(N)$ complexity. When composed with a unitary transformation $\mathbf{\Psi}$, the proximal map can be written as $\text{prox}_{\mathcal{R}\circ\mathbf{\Psi}}(\mathbf{x}) = \mathbf{\Psi}^{\mathsf{H}}\text{prox}_{\mathcal{R}}(\mathbf{\Psi}\mathbf{x})$ [133]. In this case the computational complexity mainly depends on the complexity of the transformation at hand. Hence, the computation of proximal maps only becomes an issue when more exotic regularization functions are used.

Solvers

Having setup a reconstruction problem, algorithms provided by the package *RegularizedLeastSquares.jl* can be used for its solution. The package implements many popular iterative optimization methods, including the conjugate gradients normal residual method, the Kaczmarz method, FISTA and ADMM. Importantly, the variety of the implemented solvers is sufficiently large, so that adequate solvers can be chosen depending on the reconstruction problem at hand. For example, FISTA is often used when performing CS, due to its fast convergence and the simple form of the associated proximal map. However, it is difficult to apply FISTA when $\mathcal{R} = \sum_{i=1}^{R} \mathcal{R}_i$ comprises multiple distinct regularization functions. In this case the combined proximal map associated with $\mathcal{R}$ often becomes hard to evaluate. These problems can be circumvented by using splitting-based methods, such as ADMM, which allow to reformulate the reconstruction problem in such a way that only the easy-to-compute proximal maps of the individual $\mathcal{R}_i$ need to be evaluated.

6.4 MRI Reconstruction Framework

Based on the described optimization framework, we developed a family of Julia packages, which aim to provide an image reconstruction framework for the field of MRI that is both highly performant and easily extendable. To achieve a high degree of modularity, the framework is split into seven packages each of which addresses a different aspect of the reconstruction framework. The central part of this framework is the package *MRIReco.jl*, which implements the actual image reconstruction functionality of the framework[3]. For illustration, the developed packages and their mutual dependencies are summarized in the dependency graph in Figure 6.2. An advantage of this highly modular structure is that many of the developed packages have a narrow focus and a correspondingly small set of dependencies. This opens up the possibility to only use a subset of the complete framework without unnecessarily increasing the complexity of the code to be used.

6.4.1 Functionality

The developed framework implements a range of functionalities, which have proven useful for the development, testing and application of MRI image reconstruction methods. The available functionalities include:

- *Raw Data Handling:* Support for reading data in the vendor independent ISMRMRD format [134] and in the native data format of the vendor Bruker is contained in the package *MRIFiles.jl*.
- *Imaging Operators:* Implementations of the MRI imaging operator for both Cartesian and non-Cartesian sampling schemes. Operators are available both for the idealized Fourier encoding model (4.26) and the generalized model (4.25). All of this functionality is contained in the package *MRIOperators.jl*
- *Simulation:* Methods for simulating MRI data based on software phantoms, including support for modeling B_0-inhomogeneity, R_2^*-relaxation and multiple receive coils. Moreover, basic support for the simulation of multi-echo-sequences is available. This functionality is provided by the package *MRISimulations.jl*
- *Coil Estimation:* Methods for the estimation of coil-sensitivity maps, including the sum-of-squares method and ESPIRiT [135] are provided by the package *MRICoilSensitivities.jl*.
- *Iterative Reconstruction:* Iterative reconstruction algorithms using different signal models including a single-channel model, a multi-channel model for PI-type reconstructions, and a multi-channel-multi-contrast model for the reconstruction of multi-contrast data. The methods support various regularization functions and solvers. Moreover, sampling density weights can be computed and used in order to speed up convergence of iterative solvers [44]. This functionality is contained in the package *MRIReco.jl*

[3] https://github.com/MagneticResonanceImaging/MRIReco.jl

6.4.2 Package Architecture

To guarantee modularity and accessibility, the architecture of the developed MRI reconstruction framework is based on the structure described subsection 6.2.3. As a consequence, the data flow in *MRIReco.jl* closely follows the one depicted in Figure 6.1. In the following, the central datatypes and methods of the framework will be described in more detail.

Raw Acquisition Data

Following the notation in Figure 6.1, *MRIBase.jl* provides the type `RawAcquisitionData` to describe the raw MRI data as it is stored in a file. This type is designed to be very flexible so that data from a large range of MRI experiments can be described by it. At the same time it should be structured in a way that facilitates the actual reading of raw MRI data.

To achieve this, `RawAcquisitionData` closely follows the structure of the ISMRMRD data format, which is an open source format for the storage of raw MRI data [134]. Further details about this type are summarized on the left side of Figure 6.3. An important part of it is the field `profiles`, which contains a vector with elements of the type `Profile`. Each such object describes the data measured after a single excitation during an MRI experiment. It contains the fields `head`,`traj`, and `data`, which are also used by the ISMRMRD format, to store information about the sampling trajectory and the measured data. In fact, the members of the `Profile` datatype are also bit-compatible with the corresponding HDF5 structs in ISMRMRD files.

For storing global metadata describing an experiment, the ISMRMRD uses an XML header that is extendable by custom fields. In `RawAcquisitionData`, this data is stored in the `Dict params` (i.e. an associative array). This structure provides a one-to-one mapping to the corresponding XML header.

To load measurement data into *RawAcquisitionData* objects, *MRIFiles.jl* provides a file reader for ISMRMRD files. This is a sensible choice because the ISMRMRD is an open-source format and because its developers provide converters for all major vendor formats. Hence, raw data from different imaging devices can be processed by converting the raw data file into an ISMRMRD file prior to using aforementioned ISMRMRD file reader. For convenience, *MRIReco.jl* also provides functionality to read files from the vendor Bruker. For other vendor formats, the implementation of dedicated file readers is a task for future work.

Preprocessed Acquisition Data

While `RawAcquisitionData` allows for a very flexible representation of MRI data, this generic representation can be quite inconvenient for image reconstruction. For instance, profiles are often stored in a different order than required for reconstruction. For this reason, *MRIBase.jl* introduces the datatype `AcquisitionData`, to store data in a unified way suitable for reconstruction. Its definition can be found on the right-hand side of Figure 6.3. Importantly, the k-space data is stored in the array of matrices called `kdata`. The first dimension of this array encodes the contrasts/echoes, the second dimension encodes the

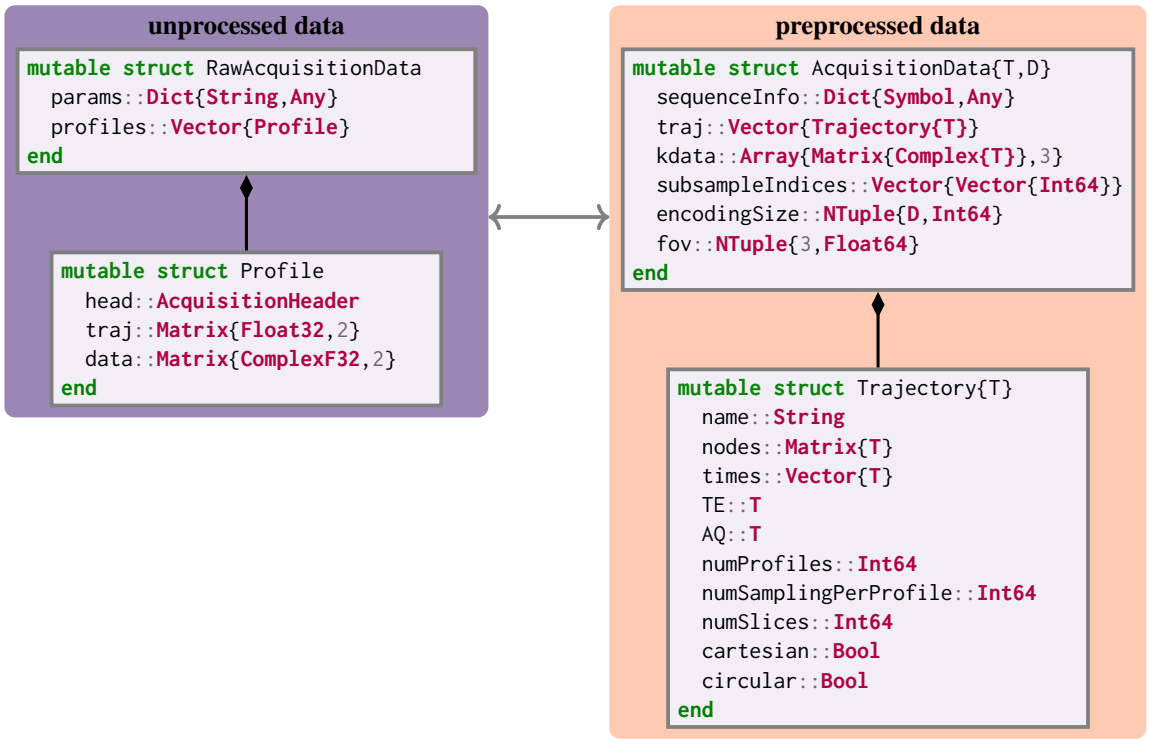


Figure 6.3: Datatypes provided by *MRIBase.jl* for the representation of MRI data. The left-hand side contains the definition of the datatype `RawAcquisitionData` used for storing unprocessed MRI data. The preprocessed MRI data is stored using the type `AcquisitionData`, whose definition is contained on the right-hand side.

slices and the third dimension encodes the repetitions acquired for each measurement. Each element of it is a matrix, whose first dimension encodes the sampled k-space nodes while the second dimension runs over all receive channels used for the acquisition. The corresponding sampling trajectory used for each contrast/echo is stored in the field `traj`, which is a vector of type `Trajectory`. Finally, the field `subsampleIndices` allows for the description of undersampled experiments, by specifying the subset of nodes of each trajectory that were actually sampled in the experiment. Finally, the encoded space is stored in the field `encodingSize`. This is especially important for non-Cartesian acquisitions where the proper size of the reconstruction grid is not clear a priori.

An important part of `AcquisitionData` objects are the `Trajectory` objects. Each such object contains the matrix `nodes`, which describes the k-space trajectory. Here the first dimension describes the spatial dimensions and the second dimension runs over the number of sampling points. Additionally, the sampling times of each node are stored in the vector `times`. This becomes important, whenever the general signal model (4.17) is used.

Image Data

After performing image reconstruction, the obtained image is returned in the form of an `AxisArray`. The latter is a special datatype, which allows to encode the dimensions of the array. As described in subsection 6.2.3, the obtained images can then be stored or processed further using tools from the Julia ecosystem.

6.4.3 Image Reconstruction

The reconstruction layer is implemented in the package *MRIReco.jl* and based on the optimization framework described in section 6.3. This frameworks provides us with a flexible way to compose image reconstruction algorithms by combining suitable solvers, operators and regularization functions.

Operators

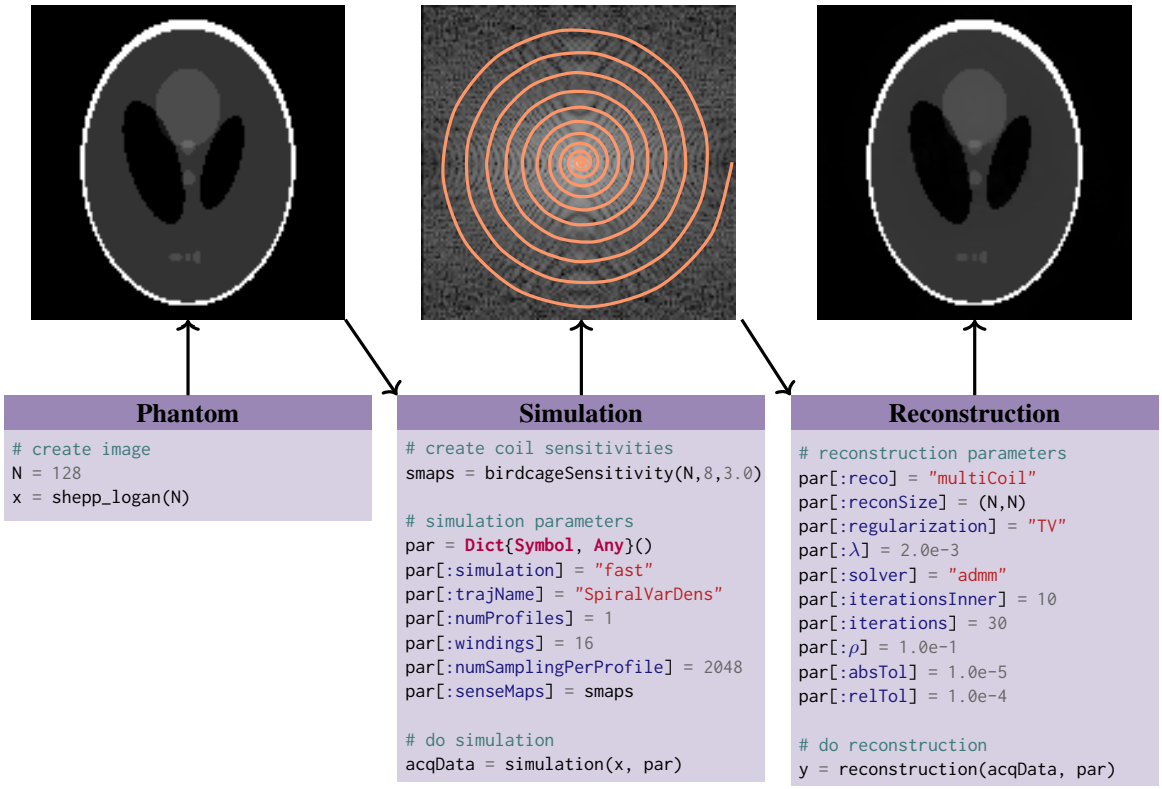
To apply the optimization framework for MRI reconstruction, the main missing piece are efficient implementations of the MRI imaging operators. For this reason, *MRIOperators.jl* provides implementations operators for the different components of the MRI imaging operator. In particular *MRIOperators.jl* contains an `FFTOp` implementing the multidimensional FFT as well as an `NFFTOp`, which implements the non-equispaced fast Fourier transform (NFFT) for multidimensional data. Finally, the `FieldmapNFFTOp` provides an implementation of the B_0-aware operator (4.25). For a more detailed discussion on these generalized Fourier-type operators, we refer the reader to the appendix A.1. To enable parallel imaging reconstructions, we additionally provide an `SensitivityOp` which multiplies an image by a set of coil-sensitivities (i.e. the operator $(\operatorname{diag}(\mathbf{p}_1)\ldots\operatorname{diag}(\mathbf{p}_L))^{\mathsf{T}}$ from subsection 4.5.2).

Making use of the described components, *MRIReco.jl* provides high-level constructors, which build the composite signal encoding operator for different reconstruction schemes. When calling a reconstruction method, these are used for setting up the given reconstruction problem.

User Interface

The interface in *MRIReco.jl* is designed so that all relevant parameters can be passed to the reconstruction method via a dictionary. The reconstruction method uses these parameters to form the main building blocks and solves the corresponding image reconstruction problem. To illustrate this, Figure 6.4 contains an example, where a set of spiral k-space data is simulated for a 128×128 pixel sized Shepp-Logan phantom and a set of 8 birdcage coil sensitivities. After simulation, image reconstruction is performed using a SENSE-type CS reconstruction with total variation regularization. The reconstruction problem is solved using ADMM with 30 iterations.

When designing the interface, an important aspect was the accessibility and extendibility of the framework. For this reason, we made sure that all major reconstruction components can be modified or exchanged using the parameter dictionary. On a high level, the overall form of the reconstruction can be



Phantom

```
# create image
N = 128
x = shepp_logan(N)
```

Simulation

```
# create coil sensitivities
smaps = birdcageSensitivity(N,8,3.0)

# simulation parameters
par = Dict{Symbol, Any}()
par[:simulation] = "fast"
par[:trajName] = "SpiralVarDens"
par[:numProfiles] = 1
par[:windings] = 16
par[:numSamplingPerProfile] = 2048
par[:senseMaps] = smaps

# do simulation
acqData = simulation(x, par)
```

Reconstruction

```
# reconstruction parameters
par[:reco] = "multiCoil"
par[:reconSize] = (N,N)
par[:regularization] = "TV"
par[:λ] = 2.0e-3
par[:solver] = "admm"
par[:iterationsInner] = 10
par[:iterations] = 30
par[:ρ] = 1.0e-1
par[:absTol] = 1.0e-5
par[:relTol] = 1.0e-4

# do reconstruction
y = reconstruction(acqData, par)
```

Figure 6.4: Example of a high-level simulation and reconstruction script using *MRIReco.jl*. The lower part of the figure contains code snippets, whereas the result of each code snippet is illustrated in the top part. In the first part a 128×128 Shepp-Logan phantom is generated (left column) and used for simulating MRI data for a spiral trajectory (middle column). Finally, a total variation regularized PI reconstruction is performed (right column).

determined by setting the parameter `:reco`. As an example, the assignment `par[:reco]="multiCoil"` in Figure 6.4 leads to a parallel imaging type reconstruction taking into account data from all coils. In contrast, the assignment `par[:reco]="standard"` leads to a reconstruction based on the single-channel model in (4.28) and `par[:reco]="multiCoilMultiEcho"` jointly recontructs data from all receive channels and all imaging contrasts contained in the acquired data. Similarly, the solver to be used can be determined by assigning its name to the parameter `:solver` in the parameter dictionary.

The regularization function to be used for image reconstruction can be determined by assigning its name to the parameter `:regularization` (e.g. `par[:regularization] = "L1"`). An alternative is to directly assign one or more `Regularization` objects to the same parameter. When working with custom regularization functions, these can easily be incorporated using the second approach.

To enable working with non-standard signals models, it is also possible to use custom implementations of the signal encoding operator. This can be achieved by building the corresponding linear operator and assigning it to the parameter `:encodingOps` in the parameter dictionary.

	Cartesian trajectory	non-Cartesian trajectory
homogeneous B_0	$\mathcal{O}(LN\log(N))$	$\mathcal{O}(L\alpha^D N\log(\alpha^D N) + Lm^D J)$
inhomogeneous B_0	$\mathcal{O}(LKN\log(N))$	$\mathcal{O}(LK\alpha^D N\log(\alpha^D N) + LKm^D J)$

Table 6.1: Computational complexity of common MRI signal encoding operators for a D-dimensional acquisition with L receive channels and J k-space nodes. Here α and m are hyperparameters of the NFFT used for non-Cartesian data. For the B_0-aware operator a number of K DFTs are used to approximate the Fourier-type operator (4.25). For further details we refer to section A.1.

6.4.4 Performance Considerations

An important requirement for the developed framework is to achieve a high computational performance on par with other state-of-the-art frameworks. This is important to ensure that the implemented reconstruction methods can be readily applied to the large datasets, which commonly occur in clinical imaging. The runtime of an image reconstruction method strongly depends on all of the specific building blocks used for its formulation, i.e. the involved linear operators, the proximal maps and the actual optimization algorithm used. For this reason, *MRIReco.jl* considers all of these with the aim of providing efficient algorithmic solutions.

Specifically in MRI, a major contribution to computational complexity is associated with the evaluation of the signal encoding operator and its adjoint. As is common practice, *MRIReco.jl* addresses this issue by providing efficient FFT-based implementations of the imaging operators, which are discussed in more detail in section A.1. These implementations lead to a loglinear computational complexity, whose exact form depends on the imaging model used and which are summarized in Table 6.1.

Another factor impacting performance is the number of iterations required by the solver at hand to converge to a high-quality result. For Cartesian k-space trajectories many optimizers show quite fast convergence because the uniform DFT is a unitary operator when using the normalization introduced in (2.5). In a gradient-based optimization method, this often leads to very efficient gradient steps which enable a fast convergence. The situation changes when dealing with non-Cartesian k-space trajectories. For this case, a popular method to accelerate convergence is to replace the reconstruction problem (6.1) by a weighted regularization problem of the form

$$\underset{\mathbf{m}}{\operatorname{argmin}} \ \|\mathbf{u} - \mathbf{S}\mathbf{m}\|_{\mathbf{W}}^2 + \mathcal{R}(\mathbf{m}). \tag{6.2}$$

Here the weighting matrix $\mathbf{W}$ contains the sampling density compensation weights introduced in (4.31). In experiments, this approach was shown to yield significant speed-ups. A drawback is that the underlying noise model no longer matches reality, which can lead to noise amplification [44]. In *MRIReco.jl*, this weighted image reconstruction scheme can be used by assigning the value `true` to the parameter `:densityWeighting` in the parameter dictionary passed to the reconstruction method.

Besides the purely algorithmic solutions presented so far, parallel code execution is another important tool to achieve fast reconstruction speeds with modern reconstruction methods. For this reason,

MRIReco.jl supports multi-threading (i.e. shared memory parallelism) and also has preliminary support for GPU acceleration, which has recently been added to the package *NFFT.jl*. To achieve decent speed-ups for a wide range of reconstruction settings, multi-threading is implemented in all the most crucial layers of the reconstruction pipeline. On a low level, several operators such as the multi-coil imaging operator and the field-map-aware imaging operator require the computation of multiple DFTs, which can be done in parallel. On a higher level, it is often desirable to perform multiple reconstructions in parallel, e.g. the reconstruction of multiple slices of a 2d acquisition. Therefore, all loops over independent reconstructions are parallelized as well. In Julia, this parallelization on different levels is possible without risking oversubscription of CPU cores with too many threads since Julia uses a thread pool allowing for nested parallelism similar to OpenMP [136], Cilk [137] and Intel TBB [138].

6.5 MPI Reconstruction Framework

This section introduces an MPI image reconstruction framework which was also developed using the described optimization tools and whose core reconstruction functionality is contained in the Julia package *MPIReco.jl*[4]. Similar to the developed MRI framework, support for the reading of raw data is capsuled away in a separate package called *MPIFiles.jl*[5] [139]. The resulting structure of the framework is also summarized in Figure 6.2.

6.5.1 Functionality

In its current form *MPIReco.jl* mainly focuses on the system-matrix-based reconstruction of MPI data. For this reason, it provides a wide range of preprocessing steps for both meausured MPI data and MPI system matrices in addition to the actual reconstruction methods. The available functionalities include:

- *Raw Data Processing:* Preprocessing methods for the raw MPI data. This includes performing averaging and performing frequency selection according to the expected SNR of the measurements. Moreover, different methods for the estimation and subtraction of background signals are implemented. These methods are supplied by the package *MPIFiles.jl*.
- *System Matrix Processing:* Methods for the processing of MPI system matrices. These include methods for forming the system matrix from the raw calibration measurements and methods for the CS-based estimation of system matrices from undersampled data. Finally, system matrices can be compressed based on their transform-domain sparsity in order to reduce their memory footprint. While the basic loading and preprocessing methods are implemented in *MPIFiles.jl*, the methods for the CS-based estimation of system matrices is part of *MPIReco.jl*.
- *Image Reconstruction:* Various system-matrix-based image reconstruction methods, including a regular single-patch reconstruction, methods for the joint reconstruction of multi-patch-data and methods for multi-contrast reconstruction. All of these are collected in the package *MPIReco.jl*.

[4] https://github.com/MagneticParticleImaging/MPIReco.jl
[5] https://github.com/MagneticParticleImaging/MPIFiles.jl

6.5.2 Package Architecture

Similar to the MRI framework, the functionality of *MPIReco.jl* reflects the scheme illustrated in Figure 6.1. However, this is made less explicit in *MPIReco.jl* because the full raw data preprocessing is performed during image reconstruction and thus no intermediate structure is formed to store the preprocessed acquisition data.

The starting point of the reconstruction pipeline is the loading of the measurement data and system matrix data, which can be done using *MPIFiles.jl*. The latter provides file readers both for the proprietary format used by the vendor Bruker and the vender independent MPI Data Format (MDF) [113]. Importantly, it provides the abstract type `MPIFile`. Objects of this type provide a file handle to the raw data files containing either measured imaging data or the measurement of an MPI system matrix. In *MPIReco.jl* this type plays a central role, as it is used to dispatch over the different available reconstruction methods.

6.5.3 Image Reconstruction

Based on the file-reading capability provided by *MPIFiles.jl*, *MPIReco.jl* provides a hierarchy of reconstruction methods each of which offers a varying degree of customizability. On the highest level, reconstruction can be performed by calling the function

```
reconstruction(fSM::AbstractString, fMeas::AbstractString; kargs...),
```

which internally performs all data loading and preprocessing besides the actual reconstruction. Here `fSM` describes the location of the `MPIFile` containing the measured system matrix and `fSM` describes the location of the corresponding measurement file. All other reconstruction parameters can be passed to the method using the the keyword arguments `kargs`. Alternatively, these parameters can be collected in a dictionary which is then passed to the reconstruction method.

On the lowest level, reconstruction can be performed by calling the function

```
reconstruction(S::Any, u::Array; kargs...).
```

Here `S` contains the system matrix to be used for image reconstruction and `u` contains the measured data to be reconstructed. Internally, this reconstruction methods uses *RegularizedLeastSquares.jl* to build a solver according to the provided keyword arguments and applies it to solve the MPI image reconstruction problem. At this level, the user has complete freedom concerning the preprocessing of the measurement data and the system matrix.

Especially for the higher-level reconstruction methods, the keyword arguments play an important role because they allow the user to specify how the reconstruction problem should be setup and what kind of preprocessing to perform. For the setup of the reconstruction problem, the keywords largely coincide with those used in `MRIReco.jl` because both make use of the package *RegularizedLeastSquares*. The other keyword arguments allow to determine important preprocessing steps, such as frequency selection or the subtraction of previously measured background signals from the measurements. For details on these parameters we refer the reader to the documentation of the package.

6.5.4 System Matrix Recovery

As discussed in chapter 5, a difference between MPI and MRI is that MPI reconstruction usually relies on a measured system matrix. Since the measurement of system matrices can take a long time, *MPIReco.jl* contains support for their recovery from undersampled data. The recovery of MPI system matrices can be performed by exploiting the transform-domain sparsity of the matrix rows using compressed sensing. In *MPIReco.jl*, the method `smRecovery` allows doing so by solving the system matrix recovery problem (5.26). We note however that `smRecovery` does not constrain the user to aforementioned sparsity-based formulation. Instead important aspects of the recovery, such as the sparsifying transform and the regularization function, can be modified by setting the appropriate keyword arguments.

6.6 Availability and Platform Support

All the Julia packages described in this work are developed within public Git repositories hosted at GitHub. The links to these repositories can be found in the previous sections dedicated to the given package of interest. The projects contain online documentation that can be accessed from the project homepages. Bug reports, feature requests and comments can be made using an issue tracker. Any commit made to the repositories is tested using continuous integration services. The packages are supposed to run on any operating system and platform that Julia itself supports. Currently, the test suite runs successfully on Linux, OS X and Windows. The software is licensed under the MIT license[6], as are most parts of Julia and its package ecosystem. The MIT licence is a permissive license and allows to use the code even in a closed-source application. We note that the developed packages only have one GPL dependency (FFTW [140]), which would need to be replaced prior inclusion into a closed source application.[7]

6.7 Experimental Evaluation

To demonstrate the utility of the presented optimization framework, we focus on the MRI reconstruction framework *MRIReco.jl*. This choice was made for multiple reasons. First of all, this reflects the fact that my own contributions mainly concerned the development of the *MRIReco.jl* and the underlying optimization framework. In contrast, *MPIReco.jl* was developed by a larger group of people with my main contribution being the development of the described methods for system matrix recovery. Secondly, *MRIReco.jl* lends itself well for this experiment as it can be compared to other established MRI reconstruction frameworks to assess its performance. This is in contrast to *MPIReco.jl*, which was the first open source MPI reconstruction framework to be published and is thus lacking candidates for such a comparison. We note however that examples for the application of *MPIReco.jl* will be shown in the

[6] https://opensource.org/licenses/MIT

[7] Massachusetts Institute of Technology (MIT) and Intel (MKL, Math Kernel Library) provide binary compatible FFTW implementations that can be used in closed-source applications.

chapters 8 and 10. Similarly, further examples for the use of *MRIReco.jl* are contained in chapters 7 and 9.

In our benchmark of *MRIReco.jl*, we apply it to openly available MRI data and perform a comparison with the popularly used C/C++ reconstruction framework BART. As a model problem for this comparison we choose an iterative SENSE reconstruction. We applied it to a radial dataset of a human brain, which was published as part of the ISMRM reproducibility challenge 1 [141]. It contains MRI measurements of a human brain acquired using a 12-channel head-coil and a radial trajectory with 96 profiles and 512 samples per profile. The coil sensitivities are estimated from the data itself using the ESIRiT [135] implementation that is part of each framework. As proposed in the reproducibility challenge, the dataset was retrospectively undersampled by reduction factors between $R = 1$ and $R = 4$. For both frameworks, images were reconstructed using an iterative ℓ_2-regularized SENSE reconstruction and the corresponding reconstruction problem was solved using the CG method. In all cases we used 20 iterations and a regularization parameter of 0.01. The reconstructions were run on a workstation equipped with two AMD EPYC 7702 CPUs running at 2.0 GHz. The data/software to reproduce this test are openly available on GitHub.[8]. Finally, we note that both of the compared frameworks are actively being developed. For this reason, the benchmark only shows a snapshot of the current performance. In particular, the benchark was performed using *MRIReco* version 0.7.1 and BART version 0.7.0.

6.8 Results

6.8.1 Runtime Performance

As discussed at the beginning of this chapter, runtime performance of a reconstruction framework is an important factor influencing its usefulness. As a matter of fact, comparing independent software frameworks can be a challenging task since frameworks often differ both in implementation details and more importantly in the choice of the implemented algorithms/optimizations and in the default reconstruction parameters. A common aspect of BART and *MRIReco.jl* is that both of them support parallel code execution in the form of multi-threading. For this reason we performed separate benchmarks using 1,2,4 and 8 threads. For each benchmark, the reconstruction is performed multiple times and the minimum time is used for comparison. This comparison guarantees that both frameworks are benchmarked under idealized but comparable conditions. In particular this procedure considers hot CPU caches.

An important aspect for the benchmarks is that BART by default evaluates the normal operator $\mathbf{S}^\mathsf{H}\mathbf{S}$ using an optimization which exploits its Toeplitz structure. This trick allows computing the action of $\mathbf{S}^\mathsf{H}\mathbf{S}$ by means of two FFTs on a grid with an oversampling factor of $\sigma = 2.0$ compared to the actual reconstruction grid. To be able to match the BART implementation, this feature was added to *MRIReco.jl* during the development of this benchmark. In addition to this Toeplitz implementation, *MRIReco.jl* allows allows evaluating the operators $\mathbf{S}$ and $\mathbf{S}^\mathsf{H}$ using NFFTs with smaller oversampling factors such as

[8] https://github.com/MagneticResonanceImaging/MRIRecoBenchmarks

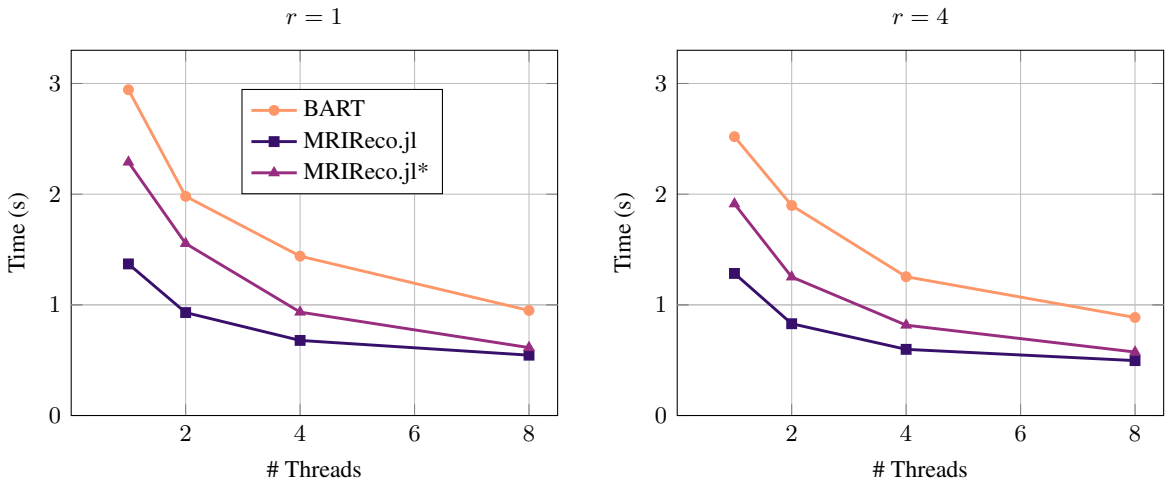


Figure 6.5: Performance comparison between BART and *MRIReco.jl*. Shown are the minimum reconstruction times of the described CG-SENSE reconstruction in dependence of the number of threads used. The left-hand side contains the results without data reduction, whereas the right contains the results for a reduction factor of 4. The circles (squares) indicate the results obtained for BART (*MRIReco.jl*) both using the Toeplitz approach. Finally, the triangles indicate the results obtained using *MRIReco.jl* without the Toeplitz approach but using an oversampling factor $\sigma = 1.25$.

$\sigma = 1.25$ (cf. appendix A.1). Theoretically, this approach may slightly reduce the accuracy of the NFFT approximation. However, in our experiments we observe that the approximation error is too small to be visually perceptible in the reconstructed images. Further details on the choice of the oversampling factor can be found in Ref. [142].

The results of the performance comparison can be seen in Figure 6.5. As one can see, both frameworks are able to perform image reconstruction within similar time scales. When considering the results for a single thread, one can see that the reconstruction time of *MRIReco.jl* is a factor $\sim$ 2 smaller than that of BART when using the Toeplitz approach. For increasing numbers of threads, the differences in reconstruction times become smaller. Nevertheless, *MRIReco.jl* remains faster than BART for all considered numbers of threads.

When comparing the Toeplitz approach with the non-Toeplitz approach, it becomes evident that the Toeplitz approach yields faster reconstructions for all undersampling factors considered. However, the observed reconstruction times become very similar for large numbers of threads. This indicates, that the NFFT benefits more strongly from parallelization than the FFTs used in the Toeplitz approach.

6.8.2 Reconstruction Accuracy

The results of image reconstruction for all considered reduction factors are collected in Figure 6.6. For *MRIReco.jl* we only show the results obtained using the Toeplitz-approach. For both frameworks only a minor decrease of image quality can be observed when the reduction factor is increased. Interestingly,

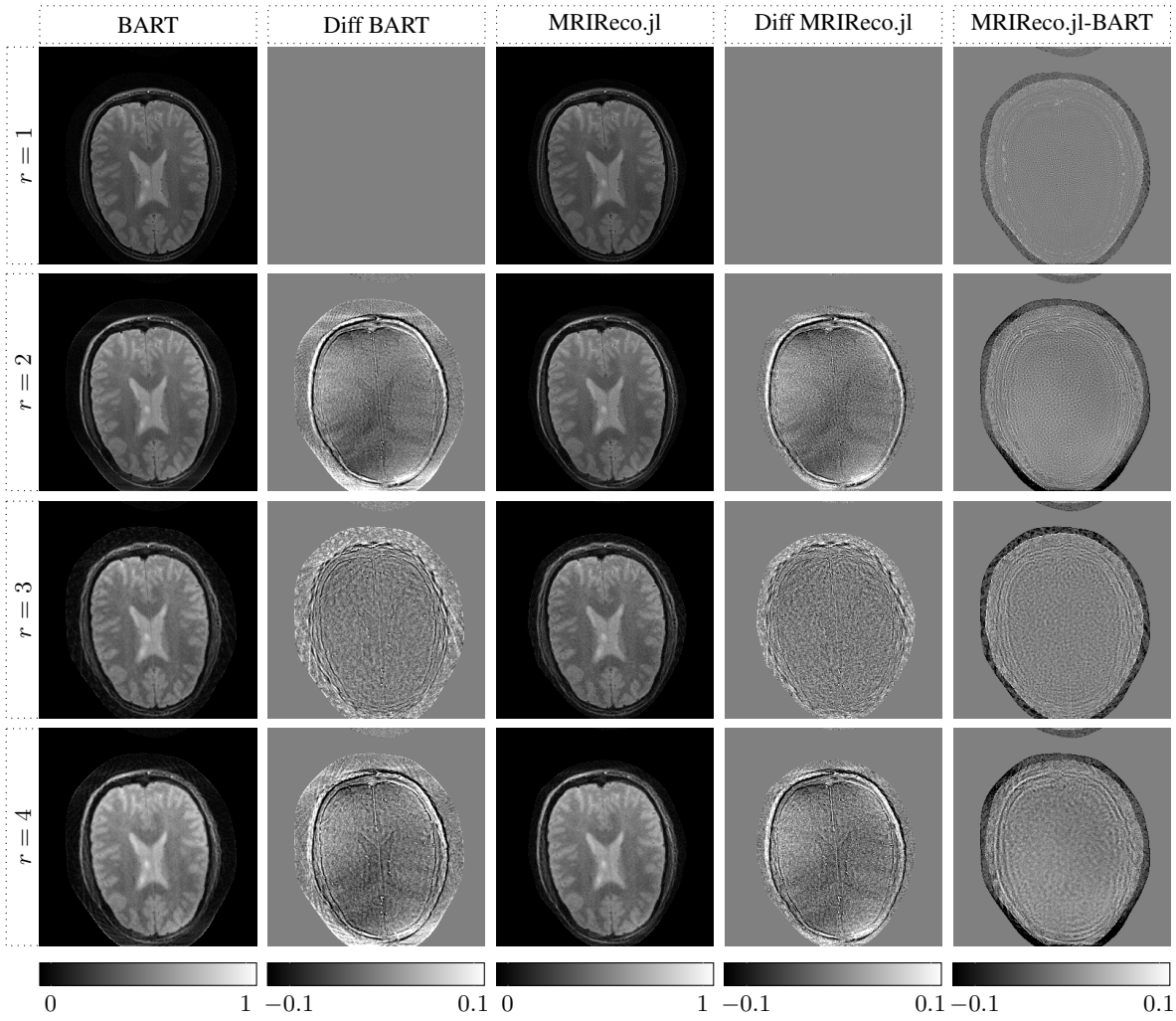


Figure 6.6: Comparison of SENSE reconstructions of the public brain dataset using BART and *MRIReco.jl*. The reconstruction results are shown in columns one and three while columns two and four show difference maps to the reference reconstruction ($r = 1$). Column five shows difference maps between the images reconstructed using both frameworks for the same reduction factor.

the reconstructions for $r = 3$ both display a smaller deviation in outer regions of the head than the even reduction factors.

When comparing the results of the two frameworks with each other one can only see negligible differences. Only the outer regions beyond the head look slightly different. These deviations are likely caused by differences in the ESPIRiT-implementation used. In fact even the difference maps compared to the $r = 1$-case look very similar. This indicates that both frameworks do implement the iterative SENSE reconstruction in a similar way. This is further supported by the difference maps between the two frameworks, which mainly show noise in the head region.

6.9 Discussion & Conclusion

The motivation underlying the development of the described Julia packages was to provide image reconstruction frameworks that simplify the implementation of new reconstruction algorithms while providing performance on par with other state-of-the-art reconstruction frameworks. A key requirement for this was the development of optimization tools that are sufficiently generic so that they can easily be incorporated into image reconstruction frameworks for different imaging modalities. In this work, such an optimization framework was realized in the programming language Julia. Based on it, the image reconstruction frameworks were developed for the imaging modalities MRI and MPI, which illustrates the usability of the developed tools.

As discussed in section 6.4 and 6.5 both frameworks feature a simple user-interface and a high level of extendability. At the same time, *MRIReco.jl* achieved a better performance than the state-of-the-art C/C++ MRI reconstruction framework BART in the benchmark discussed in section 6.8. We note however that the benchmark was performed for a very specific reconstruction algorithm (iterative SENSE) and its results are not directly applicable to other aspects of each of the frameworks.

Concerning the benchmark results it should also be highlighted that BART is based on a compiled language, whereas Julia uses just-in-time compilation. Currently, this leads to the situation that the first reconstruction run in *MRIReco.jl* takes longer than our benchmarks would suggest. This is mainly caused by the need to perform package loading and code compilation. This issue will hopefully be addressed in future versions of the language. For instance, the upcoming version Julia 1.9 will enable the caching of native machine code, which could greatly reduce the amount of compilation to be performed [143].

To obtain a highly flexible framework, modularity is an important aspect. This allows us to reuse existing software components in different frameworks. Moreover, it makes the code more maintainable as the responsibility for code is kept in the original code repositories. A challenge arising from this fine-grained modularity is a complex tree of version dependencies, which Julia addresses with its integrated package manager. Nevertheless, an important responsibility is to maintain compatibility with the depending packages. While this can prove difficult at times, we note that the Julia dependency system allows pinning packages to given versions. In this way, breaking API changes can effectively be avoided.

A design goal which was not fully addressed yet is the deployment on MRI/MPI scanners. Currently, our MRI reconstruction framework is still missing some functionality required for deployment on MRI scanners, while *MPIReco.jl* is routinely being deployed for image reconstruction at the preclinical Bruker and at the human-sized brain MPI system presented in [97]. To allow deployment of *MRIReco.jl*, one could follow the approach used in [97] and use a server written in Julia to communicate with the host server via TCP/IP. Alternatively, *MRIReco.jl* could be integrated into a framework such as Gadgetron, so that the Gadgetron infrastructure can be used for communication with the scanner. Finally, deployment is achievable via automated offline reconstruction techniques such as Yarra [144] and Autorec [145].

In summary, this chapter introduced image reconstruction frameworks for the modalities MRI and MPI. Both of them are based on a common optimization framework, whose development was a major

part of this work. Important characteristics of the developed frameworks are their modularity and open interface. This makes them particularly useful for the development and the testing of new reconstruction methods. Moreover, the frameworks are implemented purely in the programming language Julia, which allows them to reach high-performance, while simplifying the process of extending and modifying the existing code base.

7 Sparse MRI in the Presence of B_0-Inhomogeneity

This chapter investigates methods to accelerate iterative image reconstruction for MRI taking while into account the inhomogeneity of the B_0-field. The latter is a particularly challenging aspect of MRI image reconstruction because it significantly increases the numerical effort required for the evaluation of the imaging operator. This is further aggravated by the fact that the reconstruction of non-Cartesian MRI data typically converge slower than that of Cartesian acquisitions. At the same time an efficient, B_0-aware MRI reconstruction is desirable because it facilitates the adoption of highly efficient, non-Cartesian sampling trajectories in practical applications. This work was presented in [C3].

7.1 Motivation

To reduce MRI scan time as much as possible, it is desirable to combine non-Cartesian sampling trajectories with modern acquisition techniques such as PI and CS. Whereas the latter techniques reduce the number of measurements needed for reconstruction, trajectories with long readouts, such as spirals, provide a further speedup by sampling a larger portions of k-space with each excitation. The speed-up to be expected from this combination have led to a large number of studies investigating the feasibility of non-Cartesian PI for a wide range of applications including functional MRI and speech MRI [77, 78, 146].

Despite the potential of non-Cartesian trajectories, CS/PI-techniques are most often combined with Cartesian sampling schemes in practice. As discussed in chapter 4, a major reason for this is the sensitivity of long-readout trajectories to inhomogeneities of the B_0 field and the resulting artifacts if this inhomogeneity is not taken into account. To address this issue, an extended B_0-aware signal model, based on (4.25) can be used. A drawback of this approach is that the B_0-aware imaging operator no longer has the Fourier structure that is required to use efficient implementations based on the FFT. As is discussed in section A.1, this issue can be circumvented by approximating the operator using a number of K DFTs. Unfortunately, this approach still increases the numerical complexity of the imaging operator by a factor of K. A further drawback of the B_0-aware signal model is that the number of Fourier transforms needed for an accurate approximation of the imaging operator depends strongly on the field map at hand. As a

matter of fact, the most commonly used approaches do not provide a mechanism to automatically tailor this number to the given imaging conditions, which easily leads to unnecessarily long computation times.

A second challenge is the slower convergence of reconstruction methods, which is commonly observed for non-Cartesian sampling trajectories. In combination with the previously described issues, this can quickly lead to the situation that the numerical cost of image reconstruction is deemed infeasible. To accelerate convergence for non-Cartesian MRI, the most popular technique is to introduce a sampling density weighting into the image reconstruction problem, as was done in (6.2). While improving convergence, this also leads to an increased risk of noise amplification. Acceleration without compromises in the noise-model can be achieved by using suitable preconditioners with a circulant structure [80, 147, 148]. For instance, Koolsta *et al.* have proposed a circulant preconditioner to approximate the inverse of $\mathbf{S}^{\mathsf{H}}\mathbf{S}$ [57]. While showing promising results, the evaluation of these preconditioners requires additional FFTs which can be undesirable due to the associated computational cost. A simple-to-use alternative is the diagonal k-space preconditioner proposed by Ong *et al.* [43]. In their work, this preconditioner shows impressive speedups. At the same time, its diagonal structure makes it no more expensive to apply than the sampling density weighting matrix $\mathbf{W}$.

In this work, we aim to improve B_0-aware image reconstruction both on the level of the imaging operator and on the level of the optimizer. To this end, we investigate an efficient approximation of the B_0-aware imaging operator, which is based on the randomized singular value decomposition. Importantly this method allows to automatically adapt the complexity of the approximation to the field map at hand. In turn, this efficiently minimizes the number of NFFTs to be performed during image reconstruction and thus reduces computation time. As a second point, we investigate the generalization of the diagonal k-space preconditioner to the more general B_0-aware signal model (4.17). This is important, as it addresses a case that was not considered in the original paper [43]. Importantly, we seek an answer to the questions how strongly field inhomogeneities affect the structure of the preconditioner and whether it is possible to neglect them in the computation of the preconditioner.

7.2 Evaluating the Signal Encoding Operator

When performing B_0-aware MRI reconstruction, one important bottleneck is the numerical effort to evaluate the imaging operator. As is outlined in the appendix A.1, the action of the B_0-aware imaging operator is commonly computed using a low-rank approximation of the form

$$\mathbf{C} = \left(\mathrm{e}^{-\mathrm{i}\Delta\omega_0(\mathbf{r}_i)(t_j - t_E)}\right)_{j=1,\dots,J;i=1,\dots,N} \approx \mathbf{A}\mathbf{B}^{\mathsf{T}}, \tag{7.1}$$

where $\mathbf{A} \in \mathbb{C}^{J\times K}$ and $\mathbf{B} \in \mathbb{C}^{N\times K}$ and $K \ll \min(M, N)$. Having obtained appropriate matrices $\mathbf{A}$ and $\mathbf{B}$, the imaging operator can be evaluated using K NFFTs and $2K$ elementwise products, leading to a computational complexity of $\mathcal{O}\big(K\alpha^D N \log(\alpha^D N) + Km^D J\big)$, where D corresponds to the dimensionality of the encoded image. While this is a significant improvement over the $\mathcal{O}(JN)$-complexity of a

naive matrix-vector product, it is important to note that the computational effort is increased by a factor of K compared to the case of a homogeneous B_0-field.

To determine the matrices $\mathbf{A}$ and $\mathbf{B}$, the most popular approaches fix one of them to a predefined basis set and determine the remaining matrix so as to minimize the approximation error of the low-rank representation in (7.1). For instance, the time segmentation (TS) approach uses translates of the form $\mathbf{b}_{:,\kappa} = \left(\mathrm{e}^{-\mathrm{i}\Delta\omega_0(\mathbf{r}_i)\bar{t}_\kappa}\right)_{i=1,\dots,N}$ with the $(\bar{t}_\kappa)_{\kappa=1,\dots,K}$ denoting a set of fixed time points [42, 80, 81]. On the other hand, the frequency segmentation (FS) approach uses basis functions of the form $\mathbf{a}_{:\kappa} = \left(\mathrm{e}^{-\mathrm{i}\overline{\Delta\omega}_{0,\kappa}t_j}\right)_{j=1,\dots,J}$ with $(\overline{\Delta\omega}_{0,\kappa})_{\kappa=1,\dots,K}$ denoting a set of fixed frequencies [42, 149].

The just described approaches are popular for multiple reasons. On one hand, fixing one set of basis functions a priori makes it easy to optimize the remaining coefficients, thus leading to efficient algorithms for the precomputation. On the other hand, using time or frequency translates appears quite natural given the structure of the exponential matrix to be approximated. A drawback is that the resulting low-rank approximations are generally not optimal in the sense that the ℓ_2-approximation error is minimized. A second issue is that the described methods usually do not provide a simple way of determining a suitable value for the rank K. Thus, chances are high that an unnecessarily large value of K is chosen in order to avoid artifacts. This is particularly relevant for the case of 2d encoded acquisitions, where the distribution of field map values and the resulting optimal K-value can change significantly depending on the local field map of a given slice.

A way to address the first of these issues is to compute the singular value decomposition of $\mathbf{C}$. As discussed in section 2.2, this allows us to obtain the ℓ_2-optimal rank-K approximation to $\mathbf{C}$ by forming the truncated SVD

$$\mathbf{C}_K = \mathbf{U}\mathrm{diag}\left(\sigma_1, \dots, \sigma_K, 0, \dots\right)\mathbf{V}^{\mathsf{H}} \tag{7.2}$$

As a matter of fact, the large size of $\mathbf{C}$ often renders this approach infeasible. For this reason, Fessler *et al.* proposed a two-step procedure to approximate the SVD [42]. In this work, we present a modified version of this method, which allows to further reduce the precomputation time. Moreover, we show how the SVD-based method can be used to obtain an estimate for the minimum K required to achieve a given approximation error.

Determination of $\mathbf{A}$

Following our previous discussion, an optimal choice for $\mathbf{A}$ would contain the first K left-singular vectors of $\mathbf{C}$. In order to approximate these efficiently, we follow the approach in [42] and consider a histogram $\{(\Delta\omega_{0,i}, h_i)|i = 1, \dots, N_{\text{hist}}\}$ of the frequencies contained in the field map. Here $\Delta\omega_{0,i}$ denotes the field map value at the center of the i^{th} bin and h_i is the number of field map values in the corresponding bin. With this at hand we form the reduced matrix

$$\mathbf{C}_{\text{hist}} = \left(\sqrt{h_i}\mathrm{e}^{-\mathrm{i}\Delta\omega_{0,i}(t_j - t_E)}\right)_{j=1,\dots,J;i=1,\dots,N_{\text{hist}}}.$$

If N_{hist} is chosen adequately, the left singular values of this matrix provide a good approximation to those of the actual $\mathbf{C}$-matrix. At the same time, computation of the SVD is greatly simplified due to the reduced matrix size.

Beyond the histogram approach, the computation of $\mathbf{A}$ can be accelerated further based on the fact that we are only interested in the first few left-singular vectors of $\mathbf{C}_{\text{hist}}$. This allows us to make use of the randomized SVD, which uses sketching techniques to efficiently estimate a predefined number of singular vectors [150]. To do so, we introduce the parameter $K_{\max} \geq K$ defining the maximum number of singular vectors of interest and apply the randomized SVD for the computation of the corresponding singular vectors.

Determination of $\mathbf{B}$

Having computed $\mathbf{A}$, we now aim to determine a suitable $\mathbf{B}$-matrix which leads to a minimal ℓ_2-approximation error. Based on this requirement, each row $\mathbf{b}_{i,:}$ should be a solution to the problem

$$\underset{\mathbf{b}_{i,:}}{\operatorname{argmin}} \sum_{j=1}^{J} \left| \mathrm{e}^{-\mathrm{i}\Delta\omega_0(\mathbf{r}_i)t_j} - \mathbf{a}_{j,:}^{\mathsf{T}} \mathbf{b}_{i,:} \right|. \tag{7.3}$$

Especially for long-readout trajectories, the number of sampling times M greatly exceeds K. Thus, the computation of the $\mathbf{b}_{i,:}$ can be accelerated by reducing the number of time points used in the above minimization problem. Formally, this can be achieved using a histogram for the sampling times. In practice, this procedure can be simplified, because the sampling times are usually equispaced and thus have a uniform distribution. Hence, it suffices to select a sufficiently large subset $\mathcal{J} \subseteq \{1, \ldots, J\}$ of equispaced indices and solve the reduced optimization problems

$$\underset{\mathbf{b}_{i,:}}{\operatorname{argmin}} \sum_{j \in \mathcal{J}} \left| \mathrm{e}^{-\mathrm{i}\Delta\omega_0(\mathbf{r}_i)t_j} - \mathbf{a}_{j,:}^{\mathsf{T}} \mathbf{b}_{i,:} \right|. \tag{7.4}$$

Determination of K

An interesting feature of the SVD-based approach is that it allows us to estimate the minimal K necessary to achieve a given approximation error. To this end, we consider the squared approximation error of $\mathbf{C}_K$, which can be written as

$$\|\mathbf{C} - \mathbf{C}_K\|_2^2 = \sum_{k=K+1}^{\min(J,N)} \sigma_k^2.$$

Based on this expression, a practical strategy is to search for the smallest value K such that the relative approximation error is smaller than the given tolerance ε, i.e.

$$\frac{\|\mathbf{C} - \mathbf{C}_K\|_2}{\|\mathbf{C}_K\|_2} = \sqrt{\frac{\sum_{k=K+1}^{\min(J,N)} \sigma_k^2}{\sum_{k=1}^{\min(J,N)} \sigma_k^2}} \leq \varepsilon. \tag{7.5}$$

In order to use this technique with the histogram based SVD technique, we make the assumption that that the left singular vectors and singular values of $\mathbf{C}_{\text{hist}}$ are a good approximation to those of $\mathbf{C}$. Due to the construction of $\mathbf{C}_{\text{hist}}$ this assumption holds as long as N_{hist} is sufficiently large. Hence, we propose to determine K by searching for the lowest value of K fulfilling

$$\sqrt{\frac{\sum_{k=K+1}^{K_{\max}} \sigma_{\text{hist},k}^2}{\sum_{k=1}^{K_{\max}} \sigma_{\text{hist},k}^2}} \leq \varepsilon, \tag{7.6}$$

where $\sigma_{\text{hist},k}$ denotes the k^{th} singular value of $\mathbf{C}_{\text{hist}}$. To take into account only those singular values actually computed using the randomized SVD, we replaced the total number of singular values by $K_{\max}$. Here it is important to choose $K_{\max}$ sufficiently large so that the computed spectrum actually contains all non-negligible singular values. If this last assumption is violated, the computed estimates for the relative approximation error will be biased towards small values, and thus result in a too optimistic estimate for K.

7.3 Diagonal k-space Preconditionning

To solve the generic MRI reconstruction problem (6.1), most subgradient-based methods perform optimization directly in image space. This limits the range of preconditioners that can be efficiently applied to image-space operators with a sufficiently simple structure. Another promising alternative is the use of k-space preconditioners, which can be motivated by the success of the sampling density weighting approach and which were shown to enable fast convergence. To enable their use, we consider the primal dual hybrid gradient method, as proposed in Ref. [43]. The PDHG method alternatingly performs optimization in image space and in k-space according to the iteration

$$\begin{aligned} \mathbf{v}_{k+1} &= (\mathbf{1} + \sigma\mathbf{Q})^{-1}(\mathbf{v}_k + \sigma\mathbf{Q}(\mathbf{S}\bar{\mathbf{m}}_k - \mathbf{y})) \\ \mathbf{m}_{k+1} &= \text{prox}_{\tau\mathcal{R}}(\mathbf{m}_k - \tau\mathbf{S}^{\mathsf{H}}\mathbf{v}_{k+1}) \\ \bar{\mathbf{m}}_{k+1} &= \mathbf{m}_{k+1} + \Theta(\mathbf{m}_{k+1} - \mathbf{m}_k) \end{aligned} \quad . \tag{7.7}$$

Here the primal variable $\mathbf{m}_k \in \mathbb{C}^N$ contains estimates for the underlying image, whereas the dual variable $\mathbf{v}_k \in \mathbb{C}^{JL}$ resides in k-space and we assumed an acquisition of J k-space nodes with L receive channels. Moreover, $\mathbf{Q} \in \mathbb{C}^{JL\times JL}$ denotes a preconditioner, which can be used to accelerate convergence. Finally, σ and τ are step sizes used for the updates of the primal and dual variables and $\Theta \in [0, 1]$ is an extrapolation parameter which can be tuned to accelerate convergence of the algorithm [55]. For the interested reader, a derivation of the PDHG method is provided in the appendix A.2.1.

A hint to the form of the preconditioner can be obtained by inspecting the convergence condition $\sigma\tau < \lambda_{\max}(\mathbf{QSS}^{\mathsf{H}})^{-1}$ of the PDHG method [151]. Here $\lambda_{\max}$ denotes the maximum eigenvalue. To enable large step sizes, $\lambda_{\max}(\mathbf{QSS}^{\mathsf{H}})$ should be as small as possible, which can be achieved by a $\mathbf{Q}$

approximating the inverse of $\mathbf{S}\mathbf{S}^{\mathsf{H}}$. To further guarantee that $\mathbf{Q}$ can be applied easily, we constrain $\mathbf{Q}$ to be a diagonal matrix. Based on these requirements, a suitable preconditioner is given by the vector of diagonal elements $\mathbf{q}$ solving the optimization problem

$$\underset{\mathbf{q}}{\operatorname{argmin}} \left\| \operatorname{diag}(\mathbf{q})\, \mathbf{S}\mathbf{S}^{\mathsf{H}} - \mathbf{1} \right\|_2^2 . \tag{7.8}$$

A closed form solution to this problem can be derived by expanding the cost function element-wise, which yields

$$\underset{\mathbf{q}}{\operatorname{argmin}} \sum_{i=1}^{JL} \sum_{j=1}^{JL} \left| q_i \mathbf{s}_{j,:}^{\mathsf{H}} \mathbf{s}_{i,:} - \delta_{ij} \right|_2^2 .$$

Now $\mathbf{q}$ can be found by taking the derivative with respect to the components q_i setting it to zero and rearranging the corresponding optimality condition

$$q_i^{-1} = \frac{\sum_{j=1}^{JL} \left| \mathbf{s}_{i,:}^{\mathsf{H}} \mathbf{s}_{j,:} \right|^2}{\|\mathbf{s}_{i,:}\|_2^2}. \tag{7.9}$$

A direct evaluation based on this expression leads to a computational complexity of $\mathcal{O}(L^2 J^2 N)$, which is impractical for large problem sizes. Thus, we discuss efficient implementations based on the NFFT in the following subsections.

7.3.1 Preconditioner for inhomogeneous B_0-fields

For homogeneous B_0-fields, the preconditioner can be computed efficiently using NFFTs as has been shown in Ref. [43]. Here we extend the described k-space preconditioner to the case of non-negligible B_0 inhomogeneity. In this case, its computation is more challenging because the Fourier structure of the imaging operator is lost. To simplify the notation, we first restrict our consideration to the case of a single receive channel with uniform coil sensitivity. Later, we shall comment on how our results can be generalized to the multi-channel case. As an additional simplification, we present the following derivation for 1d signals, but note that the generalizations to higher dimensions is straightforward. Finally, we assume image space sampling points of the form $(r_n = -\frac{N}{2} + n - 1)_{n=1,\dots,N}$, which can easily be guaranteed by a proper choice of units.

Single-channel preconditioner

The single-channel preconditioner can be obtained by inserting the definition of the imaging operator (4.25) into (7.9). Under the described conditions, the preconditioner value corresponding to the k-space node k_i is given by

$$q_i^{-1} = \frac{1}{N} \sum_{j=1}^{J} \left| \sum_{n=-N/2}^{N/2-1} \mathrm{e}^{-2\pi \mathrm{i}(k_i - k_j)n} \mathrm{e}^{-\mathrm{i}\Delta\omega_0(n)(t_i - t_j)} \right|^2 , \tag{7.10}$$

where we additionally neglected the T_2^*-relaxation term in (4.25). To simplify the computation, we approximate the disturbance term using K frequency translates of the form $(\mathrm{e}^{-\mathrm{i}\overline{\Delta\omega}_{0,\kappa}(t_i-t_j)})_{\kappa=1,\dots,K}$. In this way, we obtain the approximation

$$\mathrm{e}^{-\mathrm{i}\Delta\omega_0(n)(t_i-t_j)} \approx \sum_{\kappa=1}^{K} b_{\kappa,n}\mathrm{e}^{-\mathrm{i}\overline{\Delta\omega}_{0,\kappa}(t_i-t_j)}$$

by following the FS approach discussed in the appendix A.1. Inserting it into (7.10) and expanding the square yields

$$q_i^{-1} = \frac{1}{N}\sum_{j=1}^{J}\sum_{m,n=-N/2}^{N/2-1}\sum_{\kappa,\kappa'=1}^{K} b_{\kappa,n}b^*_{\kappa',m}\mathrm{e}^{-2\pi\mathrm{i}(k_i-k_j)(n-m)}\mathrm{e}^{-\mathrm{i}(\overline{\Delta\omega}_{0,\kappa}-\overline{\Delta\omega}_{0,\kappa'})(t_i-t_j)}.$$

Next, we define the cross-correlation

$$\beta_{\bar{n}}^{\kappa\kappa'} = \sum_{n-m=\bar{n}} b_{\kappa,n}b^*_{\kappa',m},$$

which can be computed efficiently using a pair of FFTs. With this at hand, rearranging the sums over (m,n) allows us to write the preconditioner as

$$q_i^{-1} = \frac{1}{N}\sum_{j=1}^{J}\sum_{\kappa,\kappa'=1}^{K}\sum_{\bar{n}=-N+1}^{N-1}\beta_{\bar{n}}^{\kappa\kappa'}\mathrm{e}^{-2\pi\mathrm{i}(k_i-k_j)\bar{n}}\mathrm{e}^{-\mathrm{i}(\overline{\Delta\omega}_{0,\kappa}-\overline{\Delta\omega}_{0,\kappa'})(t_i-t_j)}.$$

To further simplify the expression, we define

$$h_{\bar{n}}^{\kappa\kappa'} = \frac{1}{\sqrt{N}}\sum_{j=1}^{J}\mathrm{e}^{\mathrm{i}(\overline{\Delta\omega}_{0,\kappa}-\overline{\Delta\omega}_{0,\kappa'})t_j}\mathrm{e}^{2\pi\mathrm{i}k_j\bar{n}},$$

which can be computed using an adjoint NFFT. Now putting everything together yields

$$q_i^{-1} = \frac{1}{\sqrt{N}}\sum_{\kappa,\kappa'=1}^{K}\mathrm{e}^{-\mathrm{i}(\overline{\Delta\omega}_{0,\kappa}-\overline{\Delta\omega}_{0,\kappa'})t_i}\sum_{\bar{n}=-N+1}^{N-1}\beta_{\bar{n}}^{\kappa\kappa'}h_{\bar{n}}^{\kappa\kappa'}\mathrm{e}^{-2\pi\mathrm{i}k_i\bar{n}}. \tag{7.11}$$

Based on this expression, the B_0-aware single-channel preconditioner can be evaluated using the scheme illustrated on the left-hand side of Figure 7.1. The main computational effort of this scheme arises from the $K(K-1)$ FFTs required to compute the cross-correlations $\beta_{\bar{n}}^{\kappa\kappa'}$, the K adjoint NFFTs for the computation of the $h_{\bar{n}}^{\kappa\kappa'}$ and the K^2 NFFTs required to evaluate the inner sums over $\bar{n}$. In summary, this leads to a computational complexity of $\mathcal{O}\big(K^2\alpha N\log(\alpha N)+K^2mJ\big)$, where α and m are the grid oversampling factor and the truncation parameter used for the computation of the NFFT. The proposed scheme thus makes it feasible to compute the preconditioner for the large image sizes commonly arising

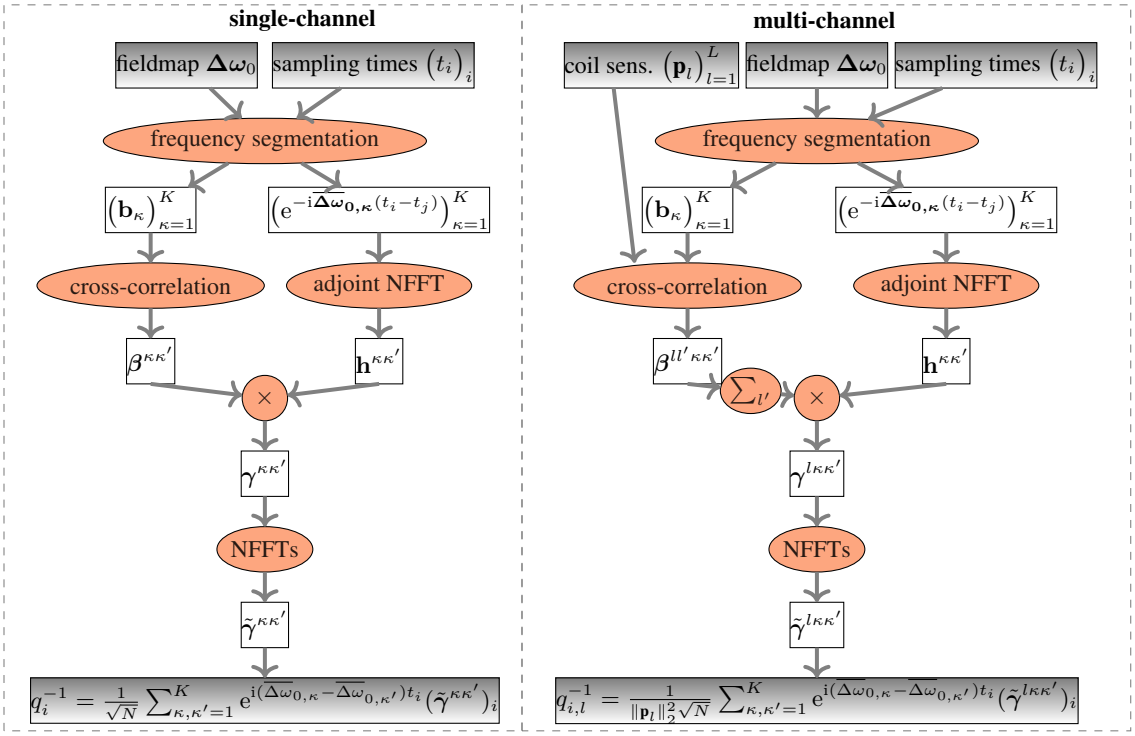


Figure 7.1: Computation of the B_0-aware ℓ_2-optimal k-space preconditioner for the single-channel case (left) and the multi-channel case (right).

in MRI. A limitation is the scaling of the computational effort with K^2. Thus, the effort increases quadratically with the level of the field inhomogeneities and the length of the readout interval.

Multi-channel preconditioner

The multi-channel preconditioner can be obtained by applying the same techniques as before to the multi-channel encoding operator (4.24). In this case an analogous calculation yields the following representation for the preconditioner

$$\begin{aligned} q_{i,l}^{-1} &= \frac{1}{\|\mathbf{p}_l\|_2^2 \sqrt{N}} \sum_{j=1}^{J} \sum_{l'=1}^{L} \left| \sum_{n=-N/2}^{N/2-1} p_l(\mathbf{r}_n) p_{l'}(\mathbf{r}_n)^* \mathrm{e}^{-2\pi\mathrm{i}(k_i-k_j)n} \mathrm{e}^{-\mathrm{i}\overline{\Delta\omega}_0(n)(t_i-t_j)} \right|^2 \\ &= \frac{1}{\|\mathbf{p}_l\|_2^2 \sqrt{N}} \sum_{j=1}^{J} \sum_{l'=1}^{L} \left| \sum_{n=-N/2}^{N/2-1} \sum_{\kappa=1}^{K} b_{\kappa,n} p_l(\mathbf{r}_n) p_{l'}(\mathbf{r}_n)^* \mathrm{e}^{-2\pi\mathrm{i}(k_i-k_j)n} \mathrm{e}^{-\mathrm{i}\overline{\Delta\omega}_{0,\kappa}(t_i-t_j)} \right|^2 \\ &= \frac{1}{\|\mathbf{p}_l\|_2^2 \sqrt{N}} \sum_{\kappa,\kappa'=1}^{K} \mathrm{e}^{-\mathrm{i}(\overline{\Delta\omega}_{0,\kappa} - \overline{\Delta\omega}_{0,\kappa'})t_i} \sum_{\bar{n}=-N+1}^{N-1} \left(\sum_{l'=1}^{L} \beta_{\bar{n}}^{ll'\kappa\kappa'} \right) h_{\bar{n}}^{\kappa\kappa'} \mathrm{e}^{-2\pi\mathrm{i}k_i\bar{n}}. \end{aligned} \tag{7.12}$$

Here we introduced the multi-channel correlation function

$$\beta_{\bar{n}}^{ll'\kappa\kappa'} = \sum_{n-m=\bar{n}} p_l(n)p_{l'}(n)^* \left(p_l(m)p_{l'}(m)^*\right)^* b_{\kappa,n} b_{\kappa',m}^*,$$

Based on (7.12), the B_0-aware multi-channel preconditioner can thus be computed with a complexity of $\mathcal{O}\big(K^2L^2\alpha N \log(\alpha N) + K^2L^2mJ\big)$ using the scheme illustrated in the right panel of Figure 7.1.

7.3.2 Preconditioner for homogeneous B_0-fields

An important special case is that of a homogeneous B_0-field, which is often assumed in MRI. In this case, the computation can be greatly simplified because it suffices to use a single translate with frequency $\overline{\Delta\omega}_{0,1} = 0$ and low-rank expansion coefficients $b_{1,n} = 1$. Inserting these simplifications into (7.12), the preconditioner becomes

$$q_{i,l}^{-1} = \frac{1}{\|\mathbf{p}_l\|_2^2 \sqrt{N}} \sum_{\bar{n}=-N+1}^{N-1} \sum_{l'=1}^{L} \rho_{\bar{n}}^{ll'} h_{\bar{n}} \mathrm{e}^{-2\pi \mathrm{i} k_i \bar{n}}, \tag{7.13}$$

where $\rho_{\bar{n}}^{ll'} = \sum_{n-m=\bar{n}} p_l(n)p_{l'}(n)^* \left(p_l(m)p_{l'}(m)^*\right)^*$ and $h_{\bar{n}} = \frac{1}{\sqrt{N}} \sum_{j=1}^{J} \mathrm{e}^{2\pi \mathrm{i} k_j \bar{n}}$. Note that this is exactly the result obtained in Ref. [43]. Thus, for homogeneous B_0, the computation of the preconditioner only involves computation of the cross-correlations $\rho_{\bar{n}}^{ll'}$ using $L(L-1)$ FFTs, computation of the $h_{\bar{n}}$ using one adjoint NFFT and evaluation of the outer sum using one NFFT. The computational complexity is correspondingly reduced to $\mathcal{O}\big(L^2 N \log(N)\big)$, thus getting rid of the quadratic scaling with K that is present in the B_0-aware case.

As a final simplification, we revisit the single-channel case. Here $\rho_{\bar{n}}^{cd}$ is constant since one assumes independent receive channels with uniform coil sensitivities $p_c(\mathbf{r}) = 1$ for all receive channels. As a result, the preconditioner can be computed even more easily with a complexity of $\mathcal{O}\big(N \log(N)\big)$. While the single-channel preconditioner can be considered suboptimal in SENSE-type parallel imaging, it is still an attractive choice because the resulting preconditioner is independent of the coil sensitivities at hand. Thus, there is no need to recompute the preconditioner whenever the coil sensitivity maps change. Moreover, the results in [43] indicate that the use of the single-channel preconditioner does not lead to significantly slower convergence than the multi-channel preconditioner.

7.4 Experiments

Synthetic Dataset

To evaluate the proposed methods, we first aim to apply them to a dataset with known ground-truth. For this purpose, we simulated an MRI dataset based on the complex-valued brain MRI images accompanying the MEDI toolbox [152, 153]. The images used as a ground-truth were obtained on a GE scanner

using a multi-echo GRE sequence with echo times $t_{E,i} = 2.8 + 5.2i\,\mathrm{ms}$ ($i = 0, ..., 8$) and an image matrix of size $256 \times 256 \times 74$. Based on this image-series, we estimated the corresponding field map by fitting a complex exponential to the signal evolution at each imaging voxel. To obtain a smooth field map with realistic values, phase unwrapping was performed subsequently. For the simulation, we further assumed a set of eight birdcage coil sensitivity maps. As a sampling trajectory we used a variable density spiral with three interleaves and a readout time of 22 ms following the analytic spiral design proposed in [154]. Using this setting, MRI measurement data was simulated by applying the corresponding MRI imaging operator (4.24) to the first echo image of the image series.

Having simulated the dataset, image reconstruction of all slices was performed using the PDHG algorithm. As a regularization function we used the ℓ_1-norm of the image coefficients in the domain of the Daubechies-2 Wavelet transform and the corresponding regularization parameter was set to a value of 0.01. To evaluate the convergence of the algorithm, we ran the algorithm for 30 iterations while monitoring the solution vector in each iteration.

One question to be addressed by our experiment is how the use of the described k-space preconditioners affects image reconstruction. For this reason, we performed image reconstruction both using the single-channel preconditioner, using the multi-channel preconditioner and without the use of a preconditioner. For the single-channel preconditioner we further considered both the B_0-aware and the B_0-unaware version of the preconditioner. Finally, we compare the reconstructions with those obtained using the FISTA method both with and without the use of sampling density compensation weights.

The second aspect to be addressed is whether image reconstruction can be accelerated by using the proposed SVD-based implementation of the imaging operator. To address this question, image reconstruction was performed both using the histogram-based time-segmentation approach as well as using the SVD-based method. For comparison we also performed image reconstruction using the standard NFFT-based signal model, which does not take into account any B_0-inhomogeneity. For the time-segmentation approach, different numbers of basis functions between 10 and 40 were tried. For the SVD-based approach, we set $K_{\max} = 40$ and used a threshold value of $\varepsilon = 10^{-4}$ to determine the number of basis functions to be used in the imaging operator.

For the synthetic dataset, all computations were performed using a single core on a laptop equipped with an AMD Ryzen 7 pro 4750u processor running at 1.7 GHz and 64 GB of RAM.

In-vivo Dataset

As a second example, we consider an *in-vivo* dataset from a functional MRI experiment that was published as a supplementary material in Ref. [146]. The dataset was measured on a 7 T ultra-high field MRI system using a single-shot spiral trajectory with an echo time of 25 ms, a readout acquisition time of 57 ms, a nominal in-plane resolution of 0.8 mm and an undersampling factor of 4. The data was acquired using a 32-channel head receive array. Conveniently, this dataset also contains the coil sensitivity maps and the field maps which were obtained during the experiment. Prior to image reconstruction, we applied geometric coil compression [83] to compress the data into a set of 8 virtual coils.

Similarly, to the synthetic dataset we performed image reconstruction of all slices using the PDHG algorithm. In this case the solution was regularized using the total variation penalty function with a regularization parameter of 0.0003. To obtain insights on the convergence behavior, we ran the solver for 50 iterations while monitoring the solution vector in each iteration.

To investigate the use of the discussed preconditioning schemes, we again repeated image reconstruction using the single-channel preconditioner, using the multi-channel preconditioner and without the use of a preconditioner. Similarly, we also performed reconstruction using the FISTA method with and without the use of sampling density compensation weights.

As for the synthetic dataset, we also performed image reconstruction using the standard NFFT-based encoding operator, using time segmentation and using the proposed SVD-based encoding operator. The TS approach was used both with 20 and with 40 basis functions. For the SVD-based imaging operator, we set $K_{\text{max}} = 40$ and used a threshold value of $\varepsilon = 10^{-4}$ to determine the number of basis functions to be used.

For the *in-vivo* dataset, all computations were performed using a single core on a workstation equipped with an Intel Xeon E5-2640 CPU running at 2.6 GHz and 500 GB of RAM.

7.5 Results

7.5.1 Preconditioner

Before examining the reconstruction results, we turn our attention to the structure of the computed k-space preconditioners. The obtained values are shown in Figure 7.2 for the first interleave and the first receive-channel of both datasets. As expected, the preconditioner has small values near the center of k-space and the values increase as one moves into the outer k-space regions. This behavior is to be expected as spiral trajectories have an increased sampling density in the center of k-space. In this sense, the preconditioner thus has a similar structure as the sampling density compensation weights $\mathbf{W}$ in (6.2). When comparing the multi-channel and the single-channel preconditioner, one finds that the former places larger weights on the measurements far from the k-space center.

A relevant question for this investigation is how strongly the inhomogeneity of the B_0-field affects the structure of the ℓ_2-optimal k-space preconditioner. Interestingly, this influence of field inhomogeneities appears to be quite small as can be seen in Figure 7.2. In fact, the differences between the single-channel and the multi-channel preconditioner appear to be much larger than those induced by field inhomogeneities. This observation is interesting because it indicates that a close to optimal performance of the k-space preconditioner can be achieved even when assuming a homogeneous B_0-field in its computation. For practical applications, this implies that the simplified expression in (7.13) can be used, which leads to a significant reduction of precomputation time.

When applied for image reconstruction, one can see that the diagonal k-space preconditioner greatly accelerates the convergence of the PDHG method. This is illustrated in Figure 7.3 where we show one reconstructed slice of the simulated dataset at different iteration numbers. As we can see, the unconditioned

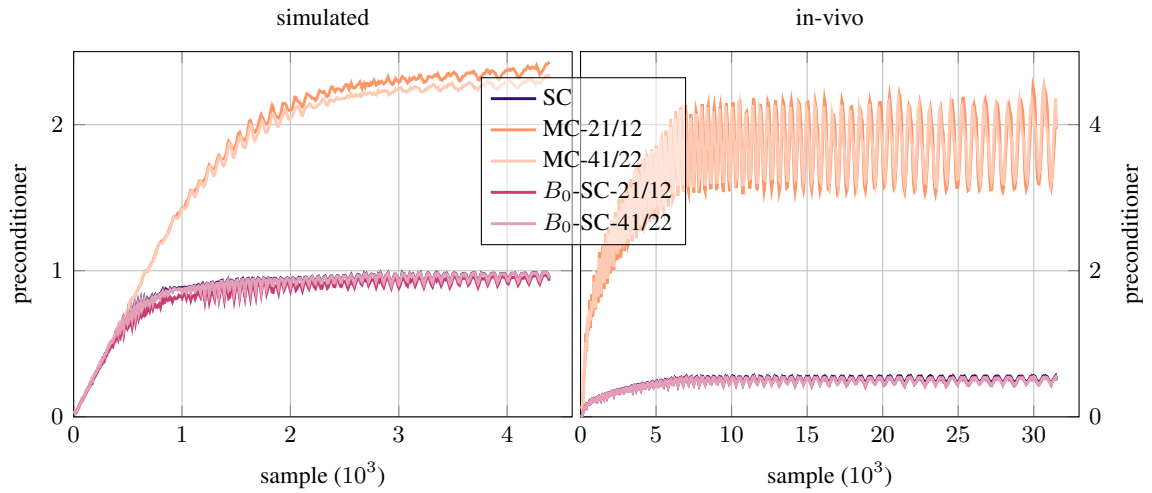


Figure 7.2: Preconditioner values for the first interleave of the spirals used in the synthetic dataset (left) and the *in-vivo* dataset (right). The results are shown for the single-channel preconditioner (SC), the multi-channel preconditioner (MC) and the B_0-aware single channel preconditioner (B_0-SC). The preconditioners are shown for the slices 21 and 41 of the simulated dataset and and for the slices 12 and 22 of the *in-vivo* dataset.

PDHG method does not yield a satisfactory result after 30 iterations, whereas the preconditioned variants yields decent reconstructions already after 10 iterations and a further improvement can be observed after 30 iterations. As expected, the results for the B_0-aware preconditioner are only negligibly different from those for the B_0-unaware preconditioner. This is a consequence of the almost identical structure of both preconditioners. Interestingly, one can also observe that the results of the single-channel preconditioner are slightly better than those of the multi-channel preconditioner in terms of the NRMSD and SSIM. This indicates a slightly faster convergence provided by the SC conditioner for the given experiment. A similar observation was also made for some of the experiments performed in Ref. [43]. Note however that this observation was not consistent across the experiments in [43]. This shows that the actual acceleration provided by the preconditioners depends significantly on the details of the MRI sequence used and the structure of the imaged object.

Similar results can also be observed for the reconstructions of the *in-vivo* dataset, which are shown in Figure 7.4. For this latter case, decent results were obtained after 30 iterations with a further improvement being visible after 50 iterations. When comparing the results to the corresponding FISTA reconstructions, we see that the preconditioned PDHG method displays a similarly fast convergence as FISTA with the use of sampling density weighting. However, we note that the fast convergence of the latter method comes at the expense of an inaccurate noise-model, which in turn can lead to noise-amplification. If the sampling density weighting is omitted, the FISTA method displays a significantly slower convergence than the preconditioned PDHG method.

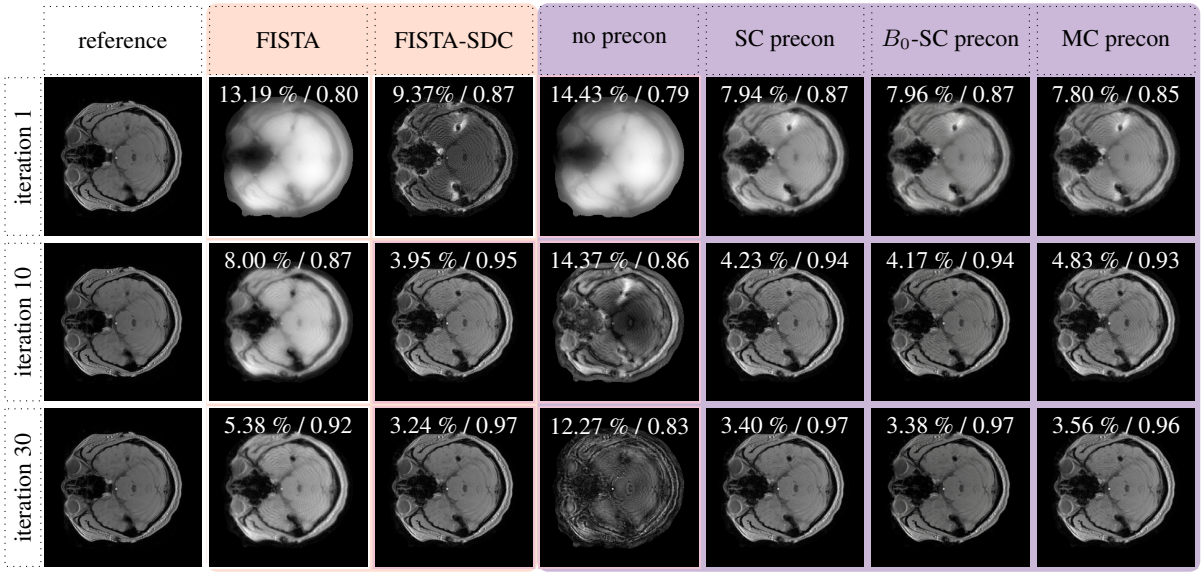


Figure 7.3: Reconstruction results for slice 21 of the simulated brain dataset. The results are shown for the PDHG method without preconditioning (fourth column), using the single-channel preconditioner (fifth column), the B_0-aware single-channel preconditioner (sixth column) and the multi-channel preconditioner (seventh column). For comparison we also show the results of FISTA reconstructions, with and without sampling density compensation, in the columns two and three. In all cases the imaging operator was evaluated using the time segmentation approach with $K = 20$ basis functions.

7.5.2 Signal Encoding Operator

A second way to accelerate image B_0-aware image reconstruction is to reduce the number of basis functions required for evaluating the B_0-aware signal encoding operator. As a matter of fact, this is no trivial task as the optimal value for K depends strongly on the local field map. This is illustrated in Figure 7.5, where we show reconstructions of the synthetic dataset for different implementations of the signal encoding operator. For both shown slices, the regular NFFT produces artifacts. However, we note that the artifacts are way more prominent in slice 21 than in slice 41. This is caused by the more severe field distortions present in slice 21. TS with 10 basis functions successfully removes the artifacts from slice 41. However, some clearly visible artifacts remain in slice 21. Finally, increasing the number of basis functions to 20 successfully removes the artifacts from both slices. A similar thing can be observed for the *in-vivo* dataset in Figure 7.6. In this case time segmentation with 20 basis functions removes most of the artifacts that are present in the NFFT-based reconstruction. However, some artifacts remain visible in slice 12. Increasing the number of basis functions to 30 removes these residual artifacts.

As described in section 7.2, the proposed SVD-based implementation of the MRI signal encoding operator provides a mechanism to automatically choose the number of basis functions, thus minimizing reconstruction time. In Figure 7.5 and Figure 7.6, it is apparent that the image quality obtained using the SVD-based operator matches that of the time-segmentation-based operator with a large number of basis functions. To investigate the computational cost, the left panel of Figure 7.7 shows the number of

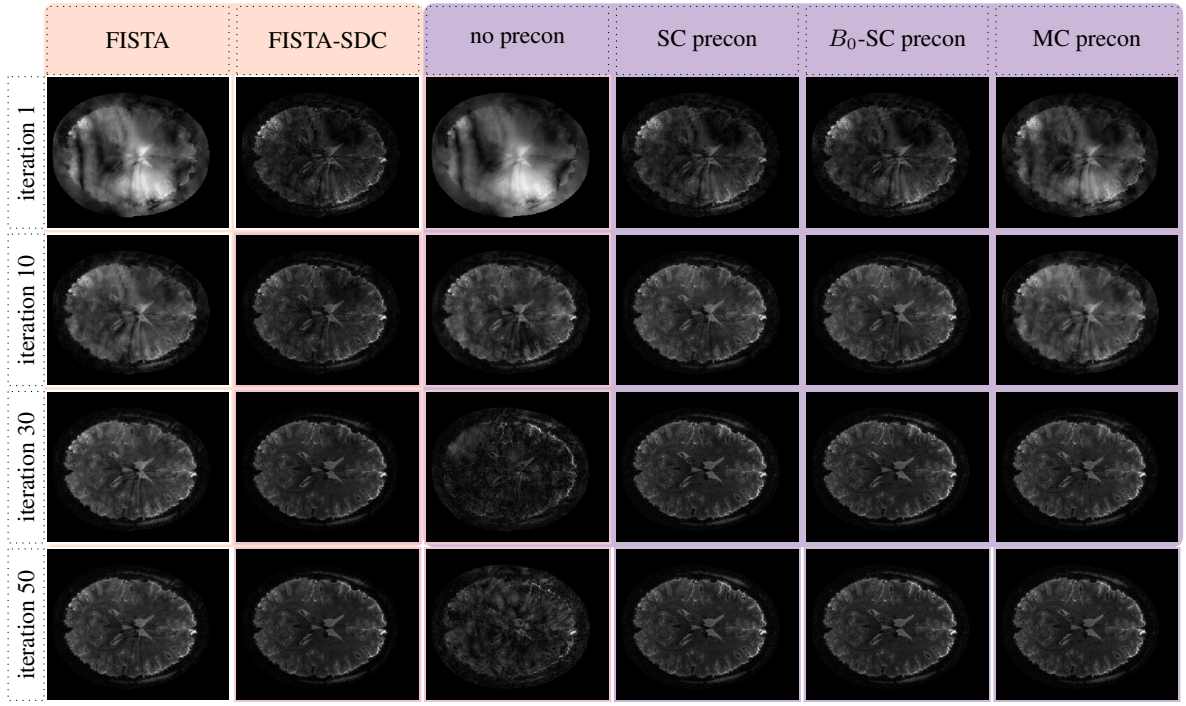


Figure 7.4: Reconstruction results for slice 22 of the *in-vivo* brain dataset. The results are shown for the PDHG method without preconditioning (fourth column), using the single-channel preconditioner (fifth column), the B_0-aware single-channel preconditioner (sixth column) and the multi-channel preconditioner (seventh column). For comparison we also show the results of FISTA reconstructions, with and without sampling density compensation, in the columns two and three. In all cases the imaging operator was evaluated using the time segmentation approach with $K = 30$ basis functions.

basis functions used by the SVD-based method in dependence of the slice to be reconstructed. This plot shows how the chosen number of basis functions varies depending on the structure of the local field map for the given slice. The average number of basis functions for the SVD-based approach was 10 for the synthetic dataset and 24 for the *in-vivo* dataset. This leads to a clear improvement over the traditional time-segmentation approach in terms of reconstruction time when assuming that a single value of K is picked such that artifacts are effectively suppressed. At the same time, the reconstruction error closely tracks that of the TS approach with a large number of basis functions. This is shown for the synthetic dataset in the right-hand panel of Figure 7.7.

A final aspect to be considered is the precomputation time needed for setting up the proposed imaging operator. For the simulated dataset, setting up the SVD-based operator took $\sim 0.6\,\mathrm{s}$ per slice. This is on par with the TS approach, for which the precomputation took $\sim 0.5\,\mathrm{s}$ per slice. Importantly, both of these times are significantly smaller than the total reconstruction time per slice. The average reconstruction time per slice was $\sim 51.9\,\mathrm{s}$ for the time-segmentation-based approach with $K = 20$ and $\sim 25.4\,\mathrm{s}$ for the SVD-based approach. For the *in-vivo* dataset the precomputation took $\sim 3.7\,\mathrm{s}$ for the SVD-based operator method and $\sim 3.5\,\mathrm{s}$ for time-segmentation with $K = 30$. Both values are small compared to the

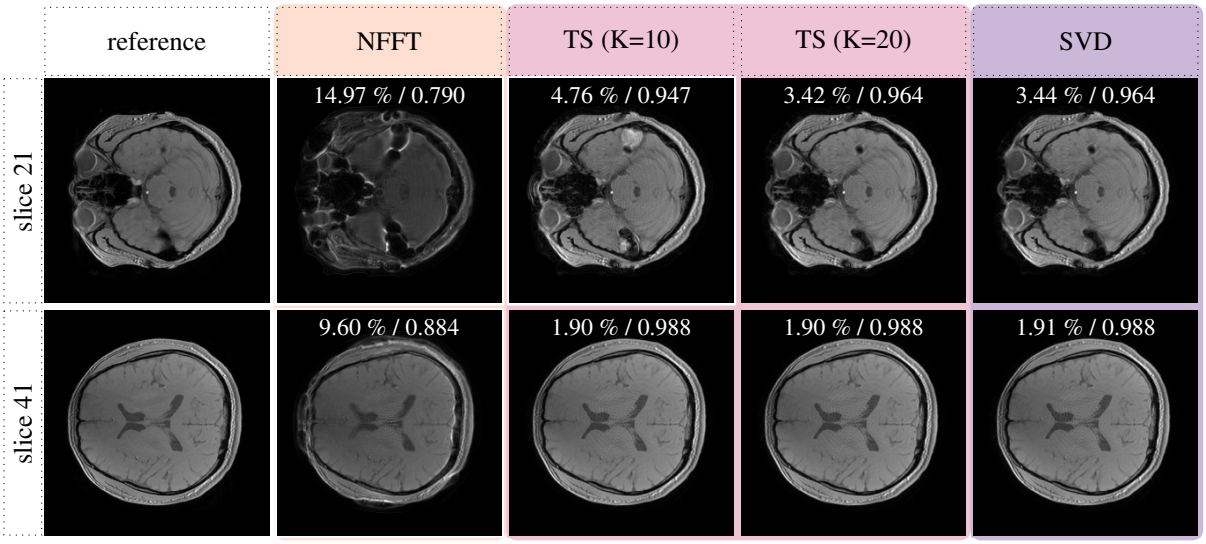


Figure 7.5: Reconstruction results for the simulated brain dataset using the PDHG method and the single-channel preconditioner. The results are shown for an NFFT-based imaging operator (column two), the time segmentation approach with 10 and 20 basis functions (columns three and four) and the proposed SVD-based method (column five).

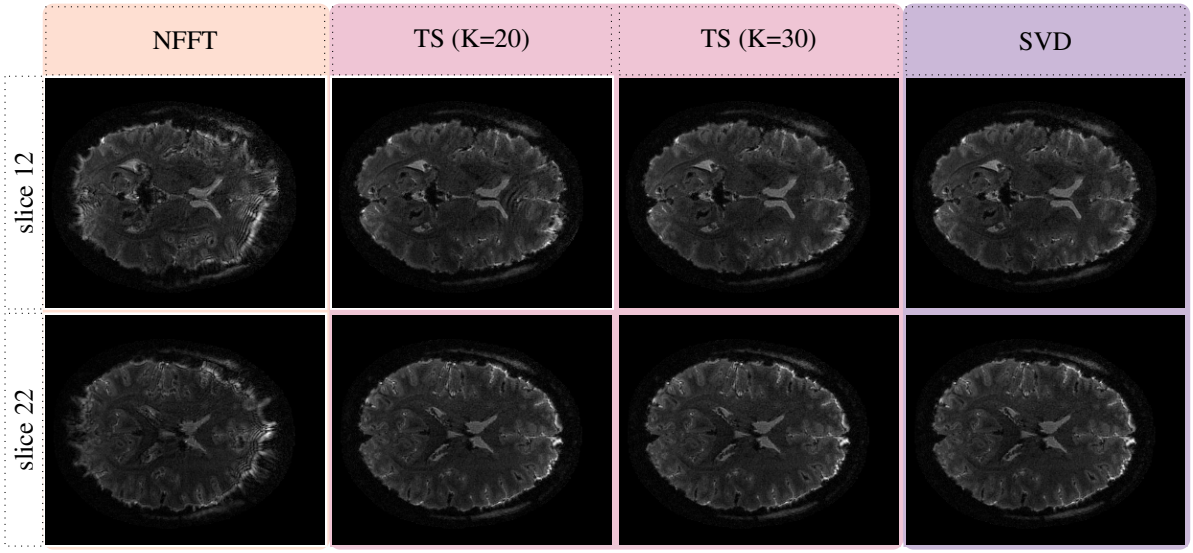


Figure 7.6: Reconstruction results for the *in-vivo* dataset using the PDHG method and the single-channel preconditioner. The results are shown for an NFFT-based imaging operator (column one), the time segmentation approach with 20 and 30 basis functions (columns two and three) and the proposed SVD-based method (column four).

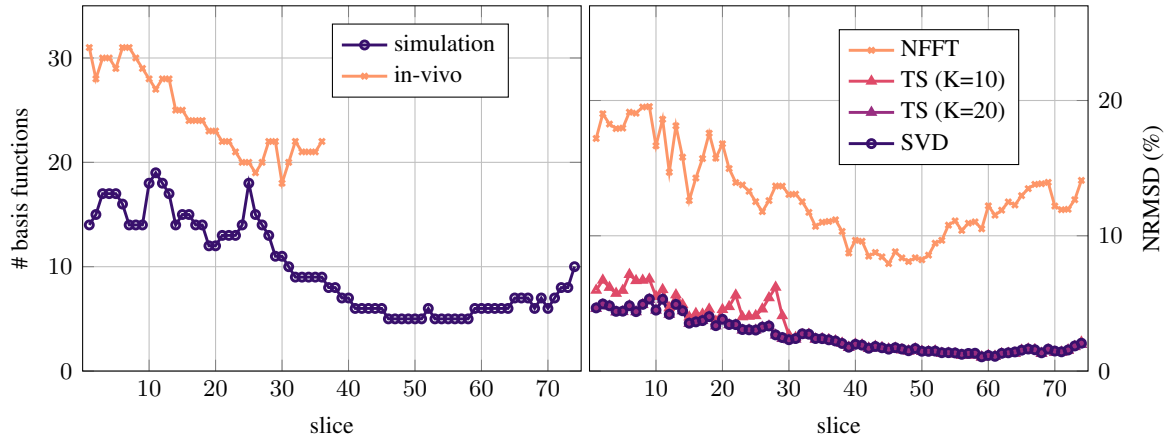


Figure 7.7: Number of basis functions used for the signal encoding operator in dependence of the slice to be reconstructed for the synthetic dataset and the *in-vivo* dataset (left panel). The right panel shows the corresponding NRMSD of the reconstructed images from the synthetic dataset.

average reconstruction time per image slice, which was $\sim 115.6\,\mathrm{s}$ for the TS approach with $K = 30$ and $\sim 99.5\,\mathrm{s}$ for the SVD-based approach.

7.6 Discussion

As shown in this work, B_0-aware MRI reconstruction can be enhanced by combining efficient approximations of the imaging operator with suitable preconditioning schemes. On the level of the imaging operator, we use an extension of the SVD-based approach originally proposed by Fessler, *et al.* In contrast to the commonly used time segmentation and frequency segmentation approaches, the SVD-based approach optimizes both factors of the low-rank approximation (7.1). Thus, it is expected to achieve lower approximation errors than the former approaches. Nevertheless, the SVD-based approach is rarely used in practice due to its comparably long precomputation time. To address this issue, our work makes use of randomized linear algebra to efficiently compute the left-singular vectors in $\mathbf{A}$ and an additional histogram-based approach to accelerate the computation of the matrix $\mathbf{B}$. In our experiments, this allowed us to compute the SVD-based approximation in a time that is only slightly larger than the time required for the commonly used time-segmentation approach [42]. Importantly, the precomputation times remain small compared to the overall reconstruction time.

An important innovation of this work is that we use the SVD-based method to automatically adapt the approximation rank to the structure of the field map. This does not only remove the need to manually tune this parameter, but it also helps to significantly reduce reconstruction time. The reason for this is that the required rank to achieve an artifact-free reconstruction varies depending on the location of the slices to be reconstructed. When setting the rank by hand, this leads to the situation that a large value

needs to be chosen for all slices in order to avoid artifacts. By adapting the rank to each image slice at hand, we avoid computing unnecessary Fourier transforms and thus minimize reconstruction time.

For completeness, we note that the NFFT-based time segmentation approach proposed by Knopp *et al.* also contains a mechanism to automatically choose the number of basis functions to be used in (7.1) [81]. As a matter of fact, the lowest rank achievable by this method is proportional to the length of the readout acquisition time and the maximum absolute value contained in the B_0-map. In a preliminary comparison, this minimum rank routinely turned out to be larger than the rank generated by the SVD-based approach. This indicates the superiority of the SVD-based approach when it comes to minimizing reconstruction time.

On the level of the image reconstruction algorithm, our results show that the preconditioned PDHG method is a useful tool for the reconstruction of non-Cartesian MRI data even in the presence of significant B_0-field inhomogeneity. In particular, the use of optimized diagonal k-space preconditioners leads to a significantly faster convergence than other methods that assume a Gaussian noise model with constant variance adequate for MRI. In our experiments, the PDHG method showed convergence speeds on par with the popular FISTA method used in conjunction with sampling density compensation. Importantly, it does so without altering the underlying noise model. In this sense, one can therefore expect the preconditioned PDHG method to be more robust especially in imaging settings with low SNR.

An important insight of our investigation is that the ℓ_2-optimal diagonal k-space preconditioner is robust to the presence of B_0-inhomogeneities. In fact, the changes induced by the field inhomogeneities are significantly smaller than those observable when transitioning from a single-channel to a multi-channel model. At the same time, our results and those in [43] show that the single and the multi-channel preconditioner lead to a similarly fast convergence of the PDHG method. Taking this into account, our results indicate that field inhomogeneities can indeed be neglected in the computation of the preconditioner, even when they have to be taken into account in the imaging operator. This is useful for practical applications because the B_0-aware preconditioner requires a significantly larger computational effort for its precomputation.

Regarding the optimization method used, this work focuses on the PDHG method. This choice is mainly motivated by the fact that it allows efficient preconditioning in k-space. A second advantage is its generality, which makes it easy to use with different regularization functions assuming that their proximal map can be computed efficiently. Beyond that, even multiple regularization functions can be easily handled by applying a suitable variable splitting scheme. This also makes the PDHG a promising candidate for other image reconstruction tasks, which benefit from the use of multiple complementary regularization terms, such as multi-contrast imaging.

In conclusion, this work has shown how non-Cartesian MRI reconstruction can be accelerated in the challenging case where B_0-field inhomogeneities cannot be neglected without inducing artifacts. Thus, it contributes to the challenge of performing efficient image reconstruction for highly accelerated non-Cartesian imaging. Both of our contributions are very generic and can be applied independently in

different reconstruction schemes. Thus, we expect them to proof useful for a wide range of acquisition schemes and image reconstruction tasks.

8

Sparse Recovery of the MPI System Matrix using Low-rank Tensors

This chapter aims to enhance existing methods for the recovery of sparsely sampled MPI system matrices. This is desirable since it helps to reduce calibration times and thus to expand the range of feasible MPI applications. To achieve this, we leverage a newly discovered representation of the MPI system matrix rows in terms of low-rank tensors. This work was published in the article [J1].

8.1 Motivation

As discussed in section 5.6, the calibration time required for the measurement of an MPI system matrix is a serious bottleneck when performing MPI experiments. This issue can be alleviated by performing an undersampled calibration measurement and using CS to recover the missing information about each system matrix row [28, 29, 119]. Following classical CS theory, previous works in this direction have focused on pseudo-random sampling patterns and exploited the sparsity of the system matrix rows when transformations such as an DFT or a DCT are applied to them. Using this approach, it has been demonstrated that high-quality system matrices can be recovered from measurements with up to 10-fold undersampling.

As has been observed in other imaging modalities, image reconstruction can often be improved by combining different forms of regularization because this better constraints the solution space [52, 155]. Similarly, it can be expected that system matrix recovery can be enhanced by combining sparsity-promoting regularization with other penalty functions that reflect our prior knowledge about the structure of the system matrix to be recovered. This motivates us to investigate other forms of prior knowledge, which can complement sparsity-promoting ℓ_1-regularization. In particular, we focus on low-rank regularization and show that the latter is a good complement to sparsity-driven methods and helps to suppress noise in the recovered frequency components.

8.2 Low-rank Property of the MPI System Matrix

8.2.1 Rank and Multilinear Rank

Before investigating the structure of the MPI system matrix, we recapitulate the concept of the rank of a matrix and discuss its generalization to higher order tensors. The latter will play an important role for the analysis of 3d system matrices as the corresponding rows can naturally be interpreted as 3-way tensors. In linear algebra, the rank of a matrix refers to the dimension of its respective row- and column space. A useful tool for its determination is the singular value decomposition, which was defined in Eq. (2.3). Importantly, it allows determining the matrix rank because the latter corresponds to the number of non-zero singular values. Moreover, for a given rank r, the ℓ_2-optimal approximation can be obtained by truncating the SVD such that only the first r singular values are kept, as done in Eq. (2.4). For a given matrix $\mathbf{M} \in \mathbb{C}^{N_1 \times N_2}$ the optimal rank-r approximation can thus be written as

$$(\mathbf{M}_r)_{i,j} = \sum_{I=1}^{r} \sum_{J=1}^{r} \operatorname{diag}(\boldsymbol{\sigma})_{I,J} \, U_{i,I} V_{j,J}. \tag{8.1}$$

In this picture, the SVD can be interpreted as a weighted sum of outer products of the singular vectors. The weighting factors for the individual summands are given by the corresponding singular values.

When dealing with higher-order tensors, the definition of the rank becomes more intricate. In fact, a variety of concepts have been proposed as generalizations of the known matrix-rank [156–158]. One popular generalization is the multilinear rank, which is derived from the higher order singular value decomposition (HOSVD) [156, 159]. For a 3-way tensor $\mathbf{T} \in \mathbb{C}^{N_1 \times N_2 \times N_3}$, the HOSVD decomposes the tensor in the form

$$T_{i,j,k} = \sum_{I=1}^{r_1} \sum_{J=1}^{r_2} \sum_{K=1}^{r_3} G_{I,J,K} \, (\mathbf{A}_1)_{i,I} \, (\mathbf{A}_2)_{j,J} \, (\mathbf{A}_3)_{k,K} \, . \tag{8.2}$$

Here the $\mathbf{A}_l \in \mathbb{C}^{N_l \times r_l}$ are unitary matrices. Moreover, $\mathbf{G} \in \mathbb{C}^{r_1 \times r_2 \times r_3}$ is a 3-way tensor. This tensor can be sliced along any of its modes, by keeping the corresponding index fixed and considering the resulting matrices. In this way one obtains r_l slices along the l^{th} mode. A characteristic of the HOSVD is that these slices of $\mathbf{G}$ are mutually orthogonal along any of the three modes.

Based on the HOSVD, the multilinear rank of $\mathbf{T}$ is defined as the tuple (r_1, r_2, r_3) that corresponds to the size of the core tensor $\mathbf{G}$. In analogy to the SVD, a low-rank approximation of $\mathbf{T}$ can be formed by truncating the core tensor appropriately. An important difference to the SVD is that the resulting approximation is not ℓ_2-optimal in general. However, it was empirically found to provide a sufficiently accurate approximation for many practical applications [159].

A related concept, which will be important for the formulation of a low-rank-promoting regularization function, is the mode-l unfolding $T_{(l)}$. It can be obtained by slicing $\mathbf{T}$ along the mode $((l+1) \mod 3)+1$ and concatenating the resulting slices, as is illustrated in Figure 8.1. An analytical expression for these

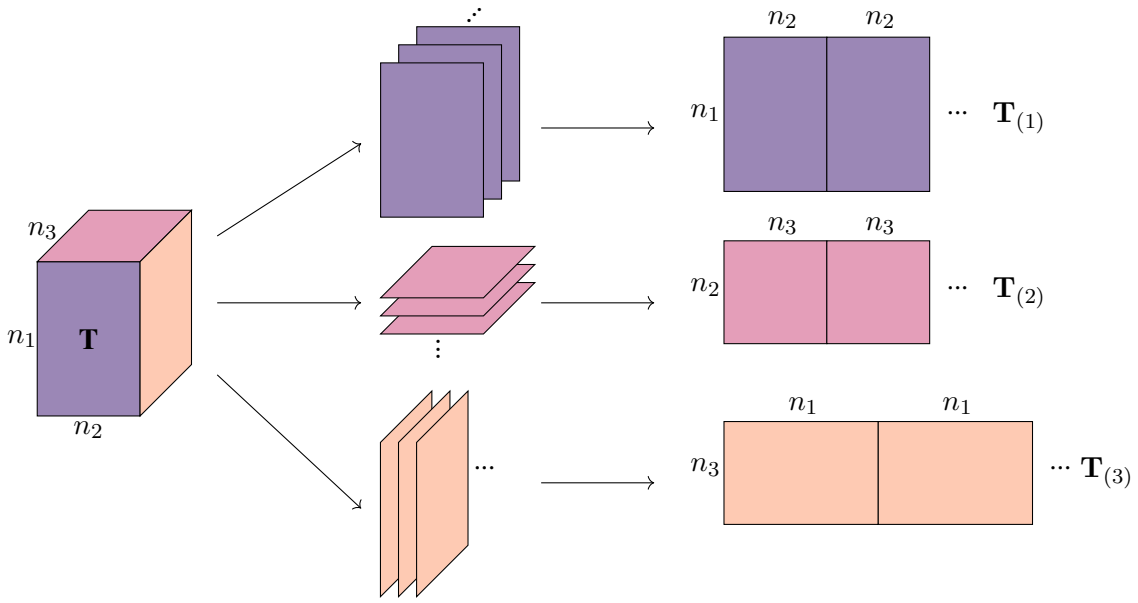


Figure 8.1: Illustration of the mode-l unfoldings of a 3-way tensor $\mathbf{T}$.

unfoldings can be readily obtained based on Eq. (8.2). For instance, the mode-1 unfolding of $\mathbf{T}$ can be written as [160]

$$\mathbf{T}_{(1)} = \mathbf{A}_1 \mathbf{G}_{(1)} \left(\mathbf{A}_3 \otimes \mathbf{A}_2\right)^{\mathsf{T}}. \tag{8.3}$$

Analogously, the other unfoldings can be obtained by cyclic permutation of the indices. Importantly, this expression implies that for a tensor with a low rank along its l^{th} mode the corresponding unfolding is a low-rank matrix. This relationship thus connects the multilinear rank of a tensor to the rank of its unfoldings. In turn, this allows us to analyze low-rank tensors using established tools for the processing of low-rank matrices.

8.2.2 Low-Rank Approximation of MPI System Matrix

Based on the previous section, we now aim to establish a connection between MPI system matrices and low-rank tensors. For this purpose, we recapitulate that the latter is usually measured on a three dimensional grid. As discussed in subsection 5.4.2, this implies that each system matrix row can equivalently be represented as a 3-way tensor whose dimensions correspond to the three spatial dimensions of the measurement grid. When we refer to the rank of a system matrix row in the following, we thus mean the (multilinear) rank of the equivalent tensor representation.

The main observation underlying the work presented in this chapter is that the rows of the system can be represented as tensors with low multilinear rank. To motivate this property, we recall that there is a strong correlation between the rows of the system matrix and tensor products of Chebyshev polynomials. As discussed in subsection 5.4.2, this correlation yields a rank-1 approximation to the system matrix

rows which indeed captures the qualitative structure of the latter fairly well. The quality of this rank-1 approximation directly motivates the idea that the system matrix rows have a low rank.

Another argument for the low rank of the system matrix rows can be constructed from the following characterization of the system matrix rows, which results from their wavelike-structure:

1. The frequency components of the system matrix are sparse in the DCT and DFT-domain [28].
2. In the DCT domain, the non-zero coefficients are contained in a small patch close to the main oscillation frequency of the given pattern [161].

Based on the second property a rank-deficient representation of a given frequency component can be obtained by constraining its non-zero DCT coefficients to aforementioned patch. In [161], the authors used this representation to remove unwanted background signals from measured system matrices. In view of these properties, it seems natural to assume that the rows of the MPI system matrix can be represented as tensors with low multilinear rank.

To illustrate that low-rank tensors indeed yield accurate approximations of measured system matrices, we first used a 2d system matrix, which is available in the Open MPI Data repository [113]. In Figure 8.2, we show the low-rank approximations which were obtained by truncating the corresponding SVD for six exemplary frequency components. One can observe that a rank-1 approximation yields rectangular patterns, which have a strong resemblance to the tensor products of Chebyshev polynomials, which were discussed in subsection 5.4.2. While this approximation captures the correct number of wave hills, it fails to describe finer structures such as the exact shape of the patterns. In contrast, the rank-6 approximation shown in the bottom row is almost indistinguishable from the measured system matrix. The visual impressions are corroborated by the NRMSD and SSIM values of the frequency components, which are superimposed onto the patterns.

If we turn to 3d system matrices, the multidimensional representation of their rows no longer has the form of a matrix but that of a 3-way tensor. In this case, low-rank approximations cannot be computed by truncating the SVD. Instead, we compute the HOSVD of the system matrix rows and truncate its core tensor. Here we need to take into account the complex-valued nature of the MPI system matrix. Thus, we apply the HOSVD separately to the real- and the imaginary parts of the given tensor.

An interesting aspect of the described low-rank approximations is the potential for denoising. In practice, measured system matrices are always corrupted by noise of varying origins. The latter can be modeled as Gaussian noise with a standard deviation that depends on the frequency component at hand. Importantly, this kind of noise cannot be represented by low-rank tensors. Thus, it is to be expected that constraining the rank of a system matrix row also has a denoising effect on the latter [162].

8.3 Sparse System Matrix Recovery

As discussed in section 5.4, the MPI system matrix is usually determined by measuring the MPI signal induced by a small sample of particles that is sequentially placed in each imaging voxel. To reduce calibration time, previous approaches have proposed to only acquire an incomplete set of measurements

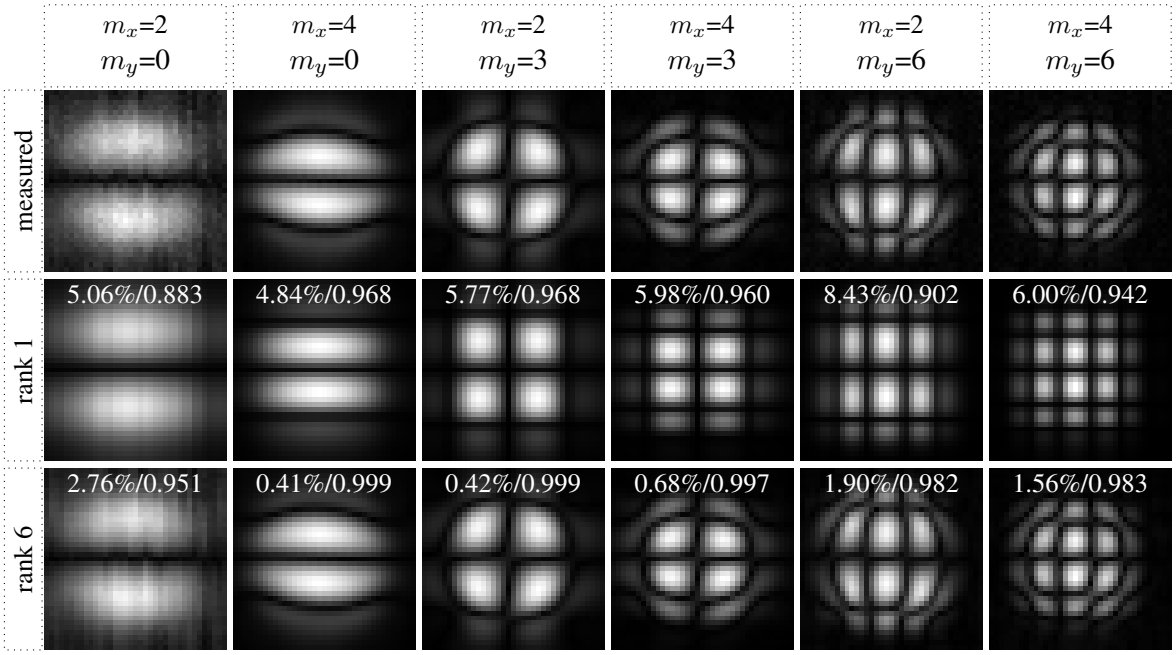


Figure 8.2: Selected representative rows from the x-channel of a 2d system matrix. The corresponding mixing factors are given on top of each column. The first row contains the measured system matrix. The second and third row show approximations of rank 1 and 6, respectively. The numbers in the patterns give their NRMSD (in %) and SSIM with respect to the measured patterns.

and to recover the system matrix row by row by solving the system matrix recovery problem (5.26), which exploits the sparsity of the system matrix rows in known domain. In this work, we follow this idea and chose the sparsifying transform to be the DCT-II, which is in line with all previous works on the CS-based recovery of MPI system matrices.

To further enhance system matrix recovery, we propose to extend problem (5.26) such that the solution is not only sparse but also has a low rank. Following this line of thought, an ideal solution would be a minimizer of the constrained optimization problem

$$\begin{aligned} \operatorname*{argmin}_{\mathbf{s}_{k,:}} \|\mathbf{\Psi}\mathbf{s}_{k,:}\|_1 \text{ subj. to } \frac{1}{2}\|\mathbf{H}_\Omega \mathbf{s}_{k,:} - \mathbf{y}_k\|_2^2 < \varepsilon^2, \\ \operatorname{mlrank}(\mathbf{R}\mathbf{s}_{k,:}) \leq (r_1, \dots, r_D). \end{aligned} \tag{8.4}$$

Here $\mathbf{R}$ is a reshaping operator, which casts the system matrix rows into their tensor representation, and $\operatorname{mlrank}(\cdot)$ denotes the multilinear rank introduced in subsection 8.2.1. A drawback of this approach is that (8.4) is a non-convex optimization problem which is hard to solve directly. For this reason, we approximate it by a convex optimization problem

$$\operatorname*{argmin}_{\mathbf{s}_{k,:}} \|\mathbf{\Psi}\mathbf{s}_{k,:}\|_1 + \mu_* \mathcal{R}_{\text{LR}}(\mathbf{s}_{k,:}) \text{ subj. to } \frac{1}{2}\|\mathbf{H}_\Omega \mathbf{s}_{k,:} - \mathbf{y}_k\|_2^2 < \varepsilon^2, \tag{8.5}$$

where $\mathcal{R}_{\mathrm{LR}}(\cdot)$ is a convex regularization function that penalizes solutions with high rank. In the following, this method will be referred to as compressed sensing with low rank (CSLR).

To enforce solutions with a low rank, the most common regularization function is the nuclear norm, which is the convex relaxation of the matrix rank. It is defined as the sum of the singular values of given matrix. Hence, for 2d system matrices, the proposed regularization function is

$$\mathcal{R}_{\mathrm{LR}}^{\mathrm{2d}}(\mathbf{x}) = \|\mathbf{R}\mathbf{x}\|_* = \|\boldsymbol{\sigma}(\mathbf{R}\mathbf{x})\|_1 \,, \tag{8.6}$$

where $\boldsymbol{\sigma}(\cdot)$ returns the singular values of its argument.

In the 3d case we also aim for $\mathcal{R}_{\mathrm{LR}}^{\mathrm{3d}}$ to be a convex relaxation of the multilinear rank. Such a convex relaxation was proposed by Signoretto *et al.* [163] and has previously shown promising results in the context of multienergy CT [160]. It is based on the observation that for a tensor with low rank in a mode l, the unfolding matrix along the same mode is a low-rank matrix. Based on this observation, the so-called tensor nuclear norm regularization function can be written as

$$\mathcal{R}_{\mathrm{LR}}^{\mathrm{3d}}(\mathbf{x}) = \sum_{l=1}^{3} \|\boldsymbol{\mathcal{U}}_l \mathbf{x}\|_* \,, \tag{8.7}$$

where the operator $\boldsymbol{\mathcal{U}}_l$ takes a vector and forms the corresponding mode-l unfolding of the associated tensor representation.

A peculiarity of the presented regularization functions is that they do not allow to directly control the rank of the solution. The latter can be controlled indirectly by the choice of the regularization parameter μ_0. However, this approach is limited because a larger regularization parameter also induces an increased amount of bias in the solution, which may counteract the benefits of the stronger rank constraints. On the other hand, our knowledge about the MPI system matrix allows us to find reasonable bounds for the rank of the system matrix rows, as has been illustrated in subsection 8.2.2. To incorporate this prior knowledge while limiting the induced bias, we also investigate the following two-step procedure, which aims to find an approximate solution to problem (8.4) with a prescribed rank $(r_1, \ldots, r_D)$ by

1. Computing the solution $\mathbf{s}_{k,:}^{\mathrm{CSLR}}$ to the CSLR problem (8.5).
2. Computing the HOSVD of $\mathbf{s}_{k,:}^{\mathrm{CSLR}}$ and truncating its core tensor according to the prescribed rank $(r_1, \ldots, r_D)$.

In the following, this two-step procedure will be referred to as compressed sensing with fixed rank (CSFR).

8.4 Solution of the CSLR problem

To solve the system matrix recovery problem (8.5), we use the Split Bregman method introduced in [52]. The latter searches for a solution to the constrained problem (8.5), by solving a series of unconstrained problems, according to the iteration

$$\begin{aligned} \mathbf{s}^{i+1} &\leftarrow \frac{1}{2}\left\|\mathbf{H}_\Omega \mathbf{s} - \mathbf{y}\right\|_2^2 + \lambda_1 \left\|\mathbf{\Psi}\mathbf{s}\right\|_1 + \lambda_* \mathcal{R}_{\mathrm{LR}}(\mathbf{s}) \\ \mathbf{y}^{i+1} &\leftarrow \mathbf{y}^i + \mathbf{y} - \mathbf{H}_\Omega \mathbf{s}^{i+1}. \end{aligned} \tag{8.8}$$

Here λ_1 and λ_* are regularization parameters for the regularization terms in the unconstrained problems. Also note that we omitted the row index k for the sake of better readability. A characteristic of the Split Bregman method is that the different regularization and data fidelity terms are effectively decoupled by the introduction of suitable auxiliary variables. This simplifies the solution of problems with multiple, non-separable regularization terms and makes it suitable for problems such as (8.5). By applying it to problem (8.5), our system matrix recovery method takes the form summarized in Algorithm 2 for the 3d case. Note that this encompasses 2d system matrices as a special case. For completeness, a derivation of the Split Bregman method is contained in the appendix A.2.2. The latter also contains an explanation of the variables in Algorithm 2. Here we content ourselves to point out that $\mathbf{x} \in \mathbb{C}^N$, $\mathbf{v} \in \mathbb{C}^N$ and $\left\{\mathbf{w}_l \in \mathbb{C}^N\right\}_{l=1,\dots,3}$ are the primal optimization variables and they are related by the constraints $\mathbf{x} = \mathbf{v}$ and $\mathbf{x} = \mathbf{w}_l$ ($l \in \{1,2,3\}$). Moreover, the variables $\mathbf{b}_v \in \mathbb{C}^N$ and $\left\{\mathbf{b}_{w,l} \in \mathbb{C}^N\right\}_{l=1,\dots,3}$ are scaled dual variables and the parameters ρ_v and ρ_w determine how strongly the constraints are enforced.

In the inner loop, the update of each primal optimization variable requires the solution of an inverse problem. Since these updates constitute the main bottlenecks of the algorithm, they will be discussed in the following. The first problem to be solved is the update of $\mathbf{x}$. By writing down the corresponding optimality condition, one obtains the linear system

$$(\mathbf{H}_\Omega^{\mathsf{H}}\mathbf{H}_\Omega + (\rho_v + 3\rho_w)\mathbf{1})\mathbf{x} = \mathbf{H}_\Omega^{\mathsf{H}}\mathbf{y}^i + \rho_v(\mathbf{v} - \mathbf{b}_v) + \rho_w \sum_{l=1}^{3}(\mathbf{w}_l - \mathbf{b}_{w,l}). \tag{8.9}$$

To efficiently solve it, we note that $\mathbf{H}_\Omega$ is a sampling matrix and thus the action of $\mathbf{H}_\Omega$ and $\mathbf{H}_\Omega^{\mathsf{H}}$ on a vector can be evaluated in $\mathcal{O}(N)$. Moreover, $\mathbf{H}_\Omega^{\mathsf{H}}\mathbf{H}_\Omega$ is diagonal and contains ones at the locations being sampled and zeros otherwise. As a result, $\mathbf{H}_\Omega^{\mathsf{H}}\mathbf{H}_\Omega + (\rho_v + 3\rho_w)\mathbf{1}$ is diagonal with strictly positive entries (as long as either ρ_v or ρ_w are positive) and (8.9) can be solved by elementwise division of the right-hand side expression by the corresponding diagonal elements.

Turning to the update of $\mathbf{v}$, the latter involves computation of the proximal map with respect to the ℓ_1-norm precomposed by $\mathbf{\Psi}$. Since our choice of $\mathbf{\Psi}$ is a unitary operator, this proximal map can be simplified to the form [133]

$$\mathrm{prox}_{\lambda\|\mathbf{\Psi}\cdot\|_1}(\mathbf{x}) = \mathbf{\Psi}^{\mathsf{H}}\mathrm{prox}_{\lambda\|\cdot\|_1}(\mathbf{\Psi}\mathbf{x}).$$

Algorithm 2: Split Bregman method for low-rank System Matrix Recovery

Input: Acquired data $\mathbf{y}_k$, undersampling operator $\mathbf{H}_\Omega$, sparsifying transform $\mathbf{\Psi}$, parameters λ_1, λ_*, ρ_v, ρ_w
Output: Recovered system matrix row $\mathbf{s}_k$
begin
  $\mathbf{y}^1 \leftarrow \mathbf{y}_k$
  $\mathbf{x}^1, \mathbf{v}^1, \mathbf{w}_l^1 \leftarrow \mathbf{H}_\Omega^\mathsf{H}\mathbf{y}$ for $l \in \{1,2,3\}$
  $\mathbf{b}_\mathbf{v}^\mathbf{1}, \mathbf{b}_{\mathbf{w,l}}^\mathbf{1} \leftarrow \mathbf{0}$ for $l \in \{1,2,3\}$
  for $i = 1, ..., I$ **do**
    for $j = 1, ..., J$ **do**
      $\mathbf{x}^{j+1} \leftarrow \underset{\mathbf{x}}{\operatorname{argmin}} \frac{1}{2}\left\|\mathbf{H}_\Omega\mathbf{x} - \mathbf{y}^i\right\|_2^2 + \frac{\rho_v}{2}\left\|\mathbf{x} + \mathbf{b}_v - \mathbf{v}\right\|_2^2 + \sum_{l=1}^{3} \frac{\rho_w}{2}\left\|\mathbf{x} + \mathbf{b}_{w,l} - \mathbf{w}_l\right\|_2^2$
      $\mathbf{v}^{j+1} \leftarrow \mathbf{\Psi}^H \operatorname{prox}_{\frac{\lambda_1}{\rho_v}\|\mathbf{\Psi}\cdot\|_1}\left(\mathbf{x}^{j+1} + \mathbf{b}_v^j\right)$
      $\mathbf{b}_v^{j+1} \leftarrow \mathbf{b}_v^j + \mathbf{x}^{j+1} - \mathbf{v}^{j+1}$
      for $l = 1, ..., 3$ **do**
        $\mathbf{w}_l^{j+1} \leftarrow \boldsymbol{\mathcal{U}}_l^\mathsf{H} \operatorname{prox}_{\frac{\lambda_*}{\rho_w}\|\cdot\|_*}\left(\boldsymbol{\mathcal{U}}_l(\mathbf{x}^{j+1} + \mathbf{b}_{w,l}^j)\right)$
        $\mathbf{b}_{w,l}^{j+1} \leftarrow \mathbf{b}_{w,l}^j + \mathbf{x}^{j+1} - \mathbf{w}_l^{j+1}$
      end
    end
    $\mathbf{y}^{i+1} \leftarrow \mathbf{y}^i + \mathbf{y} - \mathbf{H}_\Omega\mathbf{x}^{J+1}$
  end
  return $\mathbf{x}^{J+1}$
end

Conveniently, the proximal map with respect to the ℓ_1-norm can be computed by elementwise application of the soft-thresholding operator [164]

$$S_\lambda(x) = \max(|x| - \lambda, 0)\frac{x}{|x|}. \tag{8.10}$$

Based on this, the complexity of the $\mathbf{v}$-update is mainly determined by the complexity required for applying the sparsifying transform $\mathbf{\Psi}$ and its adjoint. For the case of $\mathbf{\Psi}$ being a DCT or DFT, this results in a complexity of $\mathcal{O}(N\log(N))$.

Similarly to the ℓ_1-case, the proximal map with respect to the nuclear norm can be computed by applying the soft-thresholding operator to the singular values of the argument [165]. Here the main computational cost arises from the SVD required for the computation of the singular values. For a tensor of size $N_1 \times N_1 \times N_1$, the corresponding unfolding matrices all have a size of $N_1 \times N_1^2$. Thus, computing the corresponding SVD (in economic form) has a computational complexity of $\mathcal{O}(N_1^4)$, which is only slightly superlinear [45].

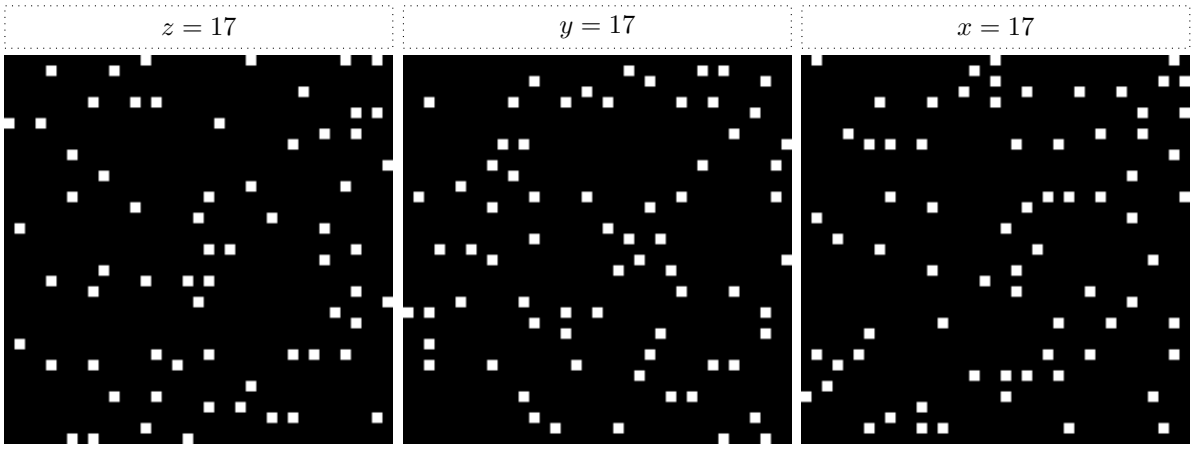


Figure 8.3: Poisson disk pattern with 20-fold undersampling. The columns show cuts in the xy-, xz-, and yz-plane respectively.

8.5 Experiments

In order to evaluate the proposed methods, we used a high-resolution 3d system matrix from the Open MPI Data repository (calibration dataset 6) [113]. The system matrix was acquired on a preclinical Bruker 15/20FF MPI scanner (Bruker Corp., Ettlingen, Germany) with a FOV of $(38.5 \times 38.5 \times 19)\,\text{mm}^3$ and a grid size of 37 x 37 x 37. The drive-field amplitudes were 12, 12, and $12\,\text{mT}/\mu_0$ and the selection field gradients were set to −1, −1, and $2\,\text{T}/(\mu_0\,\text{m})$. Finally, we only considered frequency components with an SNR ≥ 3 in our experiments.

The first experiment aims to validate the proposed low-rank model for the system matrix rows. To do so we computed approximations of the system matrix with multilinear ranks (1,1,1) to (37,37,37). Due to the symmetry of the MPI system matrix, we restricted our investigation to the special case that each mode of the tensor has the same rank.

In the second experiment we make use of the described low-rank approximation and investigate its application for sparse system matrix recovery. We thus generated undersampled data by subsampling the given system matrix using Poisson disk (PD) patterns with undersampling factors 10, 20, 30, 40 and 50. Poisson disk patterns are a popular choice in the field of CS because they provide pseudo-random sampling schemes, while avoiding excessively large areas without samples, thus leading to an increased robustness [87]. For illustration, the pattern obtained for 20-fold undersampling is shown in Figure 8.3. For each set of undersampled data, we then performed system matrix recovery using CS, CSLR and CSFR. In all cases, we used the Split Bregman method for the solution of the matrix recovery problem. The parameters associated with the ℓ_1-regularization were set to $\lambda_1 = 0.02$ and $\rho_v = 2$. These values were obtained by searching for the set of parameters minimizing the mean NRMSD of the recovered frequency components for the CS problem (5.26). For CSLR and CSFR the remaining parameters were

heuristically set to $\lambda_* = 0.02$ and $\rho_w = 0.2$. In all cases we used a maximum number of 10 outer and 50 inner iterations.

Having performed matrix recovery, the third experiment investigates the suitability of the recovered system matrix for image reconstruction. For this experiment we used a 3d dataset of a resolution phantom, which is also contained in the Open MPI data repository. The drive field amplitudes and selection field gradients for this measurement correspond to those used for the system matrix measurement. Using this data, we performed image reconstruction both using the recovered system matrices and using the original, fully sampled system matrix. Reconstruction was performed by solving the ℓ_2-regularized reconstruction problem (5.27) using the Kaczmarz algorithm with 10 iterations and a relative regularization parameter of $\lambda = 0.0025$ [121].

For a quantitative comparison of the different methods, we computed the NRMSD and the SSIM of the recovered frequency components as well as the reconstructed images. To further characterize the noise contained in the recovered frequency components, we computed estimates for the noise-to-signal ratio (NSR) as is outlined in the following. For the estimation of the NSR, we consider the outermost $3 \times 3 \times 3$ voxels of each corner in the FOV and we let $\mathbf{s}_{k,:}^{\text{bg}}$ contain the values of $\mathbf{s}_{k,:}$ in these voxels. Since the FOV is significantly larger than the DF-FOV (which is $(20 \times 20 \times 10)\,\text{mm}^3$)), one can reasonably assume constant values for the system matrix in these corners [166]. Based on this, we compute the NSR as

$$\text{NSR}(\mathbf{s}_{k,:}) = \frac{\left\| \mathbf{s}_{k,:}^{\text{bg}} - \bar{\mathbf{s}}_{k,:}^{\text{bg}} \right\|_2}{\sqrt{\left\| \mathbf{s}_{k,:}^{\text{bg}} \right\|_0} \left\| \mathbf{s}_{k,:} \right\|_\infty}. \tag{8.11}$$

Here each entry of $\bar{\mathbf{s}}_{k,:}^{\text{bg}}$ contains the mean value of the corresponding corners from $\mathbf{s}_{k,:}^{\text{bg}}$. Note that the NSR is defined completely analogously to the NRMSD, albeit giving an estimate for the noise level instead of for the similarity between signals. In analogy to the $\overline{\text{NRMSD}}$ and $\overline{\text{SSIM}}$, we also use the mean NSR as a measure for the noise contained in a full system matrix and denote the former by $\overline{\text{NSR}}$.

8.6 Results

8.6.1 Low Rank of the System Matrix Rows

The results of our first experiment clearly demonstrate that system matrices can be approximated well by constraining their multilinear rank. As an example, Figure 8.4 shows approximations with multilinear ranks $(1, 1, 1)$ and $(6, 6, 6)$ for a selection of frequency components. For a multilinear rank of $(6, 6, 6)$, the obtained approximations are almost indistinguishable from the reference patterns. The only perceivable differences are caused by the noise contained in the measured system matrix. A similar conclusion can be drawn from Figure 8.5, which shows histograms of the NRMSD of the low-rank approximations for different multilinear ranks. For a multilinear rank of (15,15,15) one can observe that most frequency components are approximated with a high accuracy. Moreover, the corresponding NSR plot reveals that

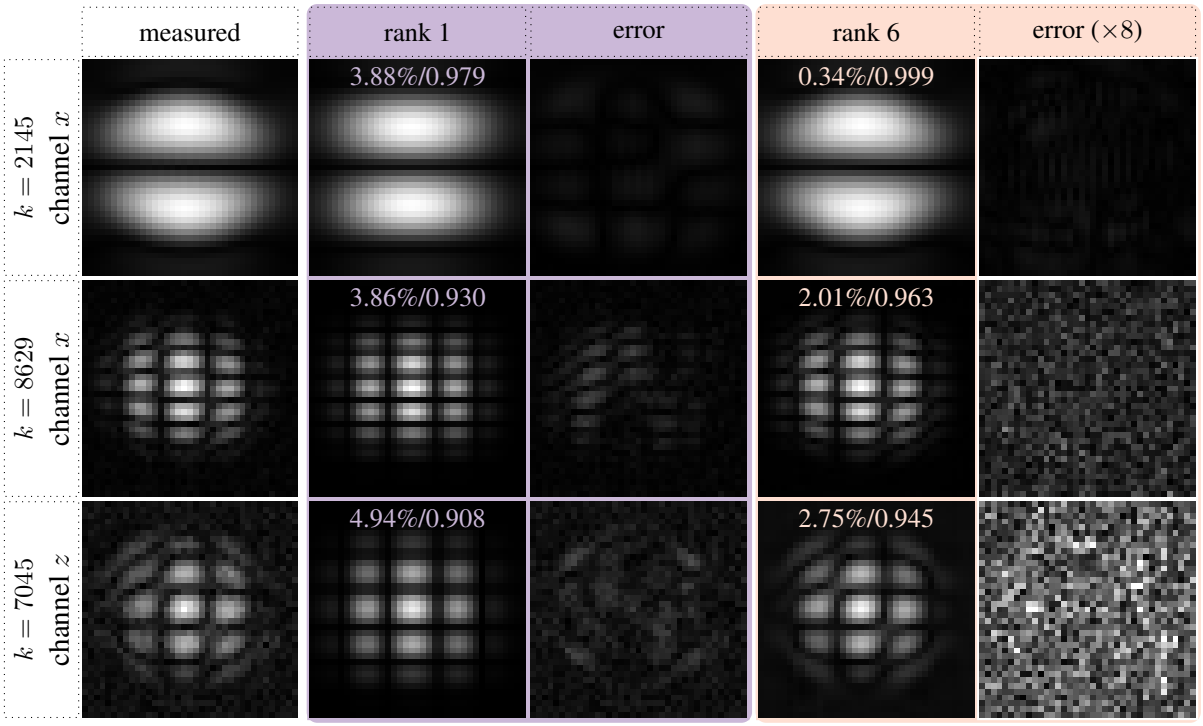


Figure 8.4: Selected rows of a 3d system matrix. Here the label k denotes the frequency index of the displayed rows. The first column shows the measured system matrix. Columns two and four show approximations of multirank $(1, 1, 1)$ and $(6, 6, 6)$ respectively. The third and fifth row show the corresponding error plots. Each plot shows the 17^{th} slice in the xy-plane. The overlayed numbers give the NRMSD (in %) and the SSIM of the corresponding patterns with respect to the measured system matrix. Note that the error plot for the rank-6 approximation was scaled by a factor of 8 to increase the contrast.

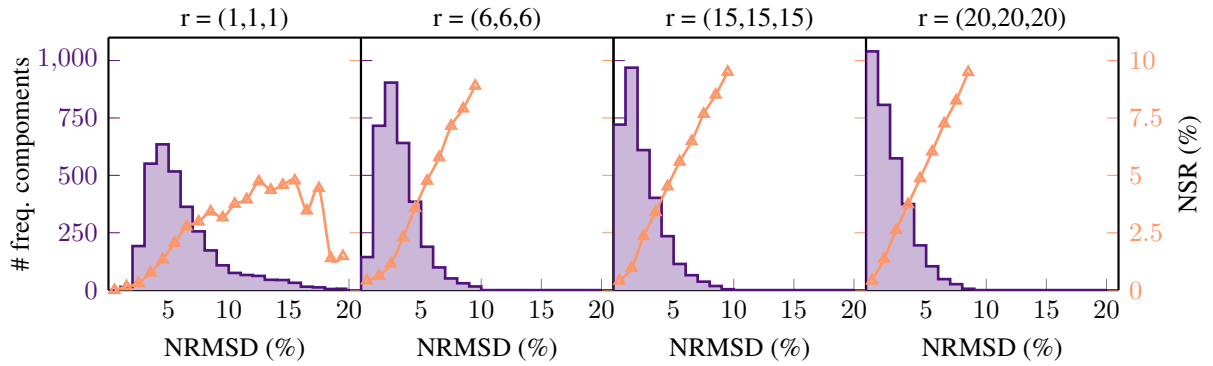


Figure 8.5: Low-rank approximation error for the 3d system matrix and multilinear ranks (1,1,1), (6,6,6), (15,15,15) and (20,20,20). The purple plot shows histograms of the NRMSD of the approximated frequency components. The orange line shows the mean NSR of the measured frequency components in the respective bin.

the patterns with a comparably large approximation error mainly have a low SNR. Hence, for these patterns a large portion of the approximation error can be attributed to measurement noise. Similar observations can be made for a multilinear rank of $(6, 6, 6)$ although the error distribution is slightly shifted towards larger values due to the more rigorous rank truncation performed.

An important question is to which extend the rank constraint leads to a removal of noise from the signal and to which extend a bias is introduced. Our results indicate that for an adequately chosen rank, a denoising can be achieved while introducing only a small amount of bias. In the difference images between the measured system matrix and the low-rank approximations, denoising leads to a noise-like image, whereas a bias manifests itself in more structured artifacts. As can be seen in Figure 8.4, the rank-1 approximation clearly induces a bias, which is reflected in the corresponding error maps. In contrast, the error maps for the rank-6 approximation are rather noise-like. Only for the 2145^{th} frequency component, a slight bias can be seen. This is understandable due to the high SNR of that frequency component. This trade-off between noise and bias is also reflected in Figure 8.5. For $r = (1, 1, 1)$, the error distribution is clearly shifted to larger values compared to the higher-rank approximations, which indicates the induced bias. For the higher-rank approximations, the error distribution changes only very slightly. Moreover, the bins with high NRMSD appear to be contain many frequency components with low SNR. This illustrates that noise is removed, while only a small amount of bias is induced.

8.6.2 System Matrix Recovery

To assess the performance of the proposed system matrix recovery methods, we first consider the case of 20-fold undersampling. In this case, the results of our experiment show an overall improvement of the low-rank methods over standard sparsity-based CS methods. This is clearly reflected in the $\overline{\text{NRMSD}}$, $\overline{\text{SSIM}}$ and the number of frequency components with NRMSD smaller than 5%. Whereas the $\overline{\text{NRMSD}}$ and $\overline{\text{SSIM}}$ give us aggregate measures for the similarity between the fully-sampled and the recovered system matrices, the last quantity can be viewed as a measure for the number of successfully recovered frequency components. For the different methods, the respective $\overline{\text{NRMSD}}$-values are 5.39% (CS), 4.89% (CSLR) and 4.73% (CSFR). The corresponding $\overline{\text{SSIM}}$-values are 0.841 (CS), 0.874 (CSLR) and 0.883 (CSFR). Finally, the number of frequency components with NRMSD smaller than 5% are 1570 (CS), 1882 (CSLR) and 1985 (CSFR). To gain a visual impression of the recovered system matrices, a selection of the recovered frequency components is shown in Figure 8.6. As one can see, the 2145^{th} frequency component of the x-channel is recovered with a high accuracy by all methods. For the other frequency components, CSLR yields a result with better quality and less perceivable noise than the CS recovery. For CSFR, the quality of the recovered patterns is increased even further.

Importantly, our results indicate that the benefits of the low-rank approximation (see subsection 8.6.1) carry over to the system matrices recovered using CSLR and CSFR. For instance the increased robustness of CSLR and CSFR against measurement noise can be attributed to the denoising associated with forming a low-rank approximation. In Figure 8.6, this effect is most visible for the 7045^{th} frequency component of the z-channel. Whereas the CS recovery of this pattern clearly suffers from noise-amplification, the

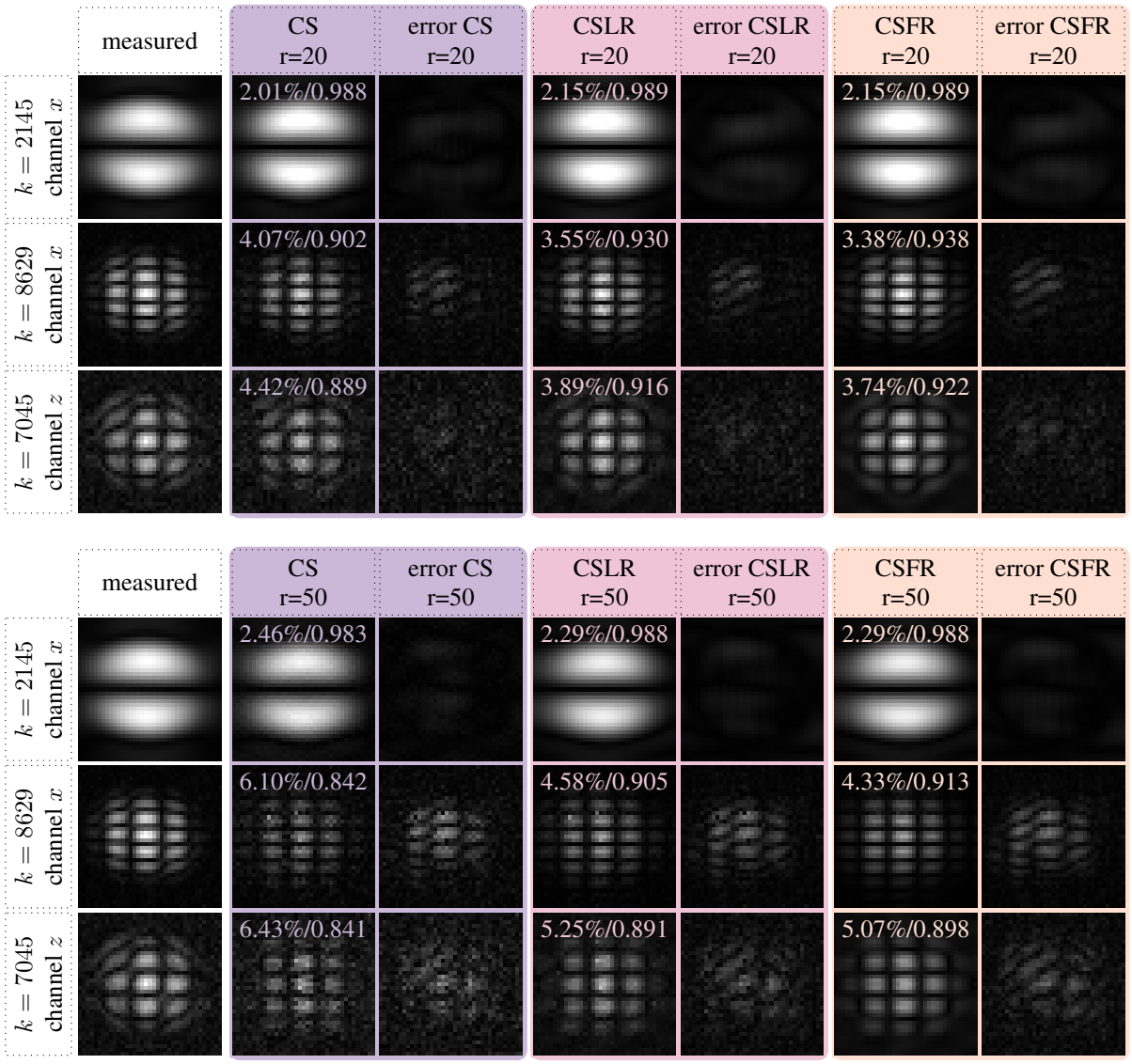


Figure 8.6: Selected rows of the 3d system matrices recovered from 20-fold undersampled data (top) and 50-fold undersampled data (bottom). Here the label k denotes the frequency index of the displayed rows. The first column contains the measured system matrix, whereas the second, fourth and sixth column show the results obtained with CS, CSLR and CSFR, respectively. Columns 3, 5 and 7 show the corresponding error plots. All shown slices correspond to the 17^{th} slice in the xy-plane. The overlayed numbers give the NRMSD (in %) and the SSIM of the corresponding patterns with respect to the measured system matrix.

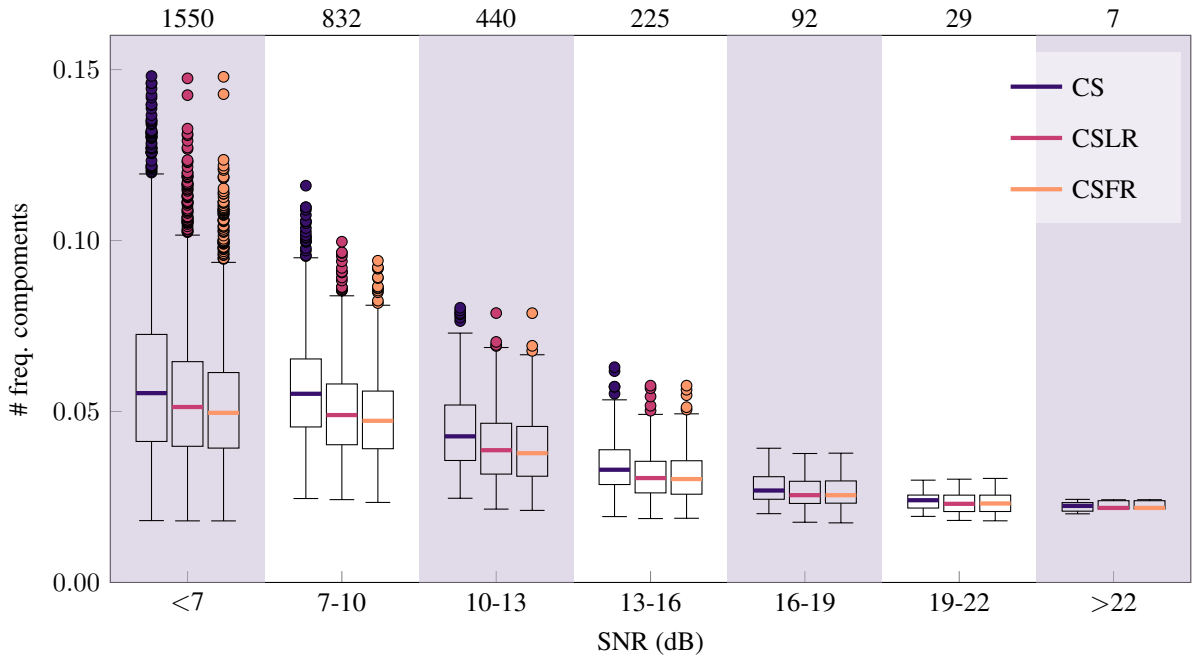


Figure 8.7: Box-Whiskers plot showing the NRMSD of the recovered system matrices obtained using CS, CSLR and CSFR. The boxes accumulate the 25%- to 75%-quantile. The numbers on the x-axis denote the range of SNR values (in dB) contained in each bin. The numbers on top of the SNR bins denote the number of frequency components contained in the latter.

additional noise is reduced in the CSLR recovery. For the CSFR recovery, noise-like artifacts are not visible at all. The described robustness against noise becomes even more apparent when considering the error maps in Figure 8.6. Whereas the error of the CS recovery is dominated by a noise-like behavior, the error maps for the low-rank methods show more structured artifacts. Note however that no additional type of artifacts seem to be introduced as the same kinds of structures can also be seen in the error maps of the CS recovery. Despite the apparent advantages of CSLR and CSFR, CS seems to yield smaller recovery errors for the frequency components with high SNR. In Figure 8.6, this can be observed for the frequency component 2145. This behavior is to be expected and it can be explained by the bias introduced by the additional regularization terms of CSLR and CSFR. Importantly, the described effects are not tied to specific frequency components but can also be observed for the full system matrix. This is illustrated in the Box-Whiskers plot in Figure 8.7. The plot clearly shows the improvement of CSLR and CSFR for frequency components with low SNR, while all three methods perform comparably well for frequency components with high SNR. To put this into perspective we note that bins with low SNR actually contain the largest number of frequency components.

For practical applications, it is important how well the recovered frequency components perform when used for image reconstruction. To answer this question, the images reconstructed using the recovered system matrices are shown in Figure 8.8. As can be seen, all reconstructions accurately recover the shape

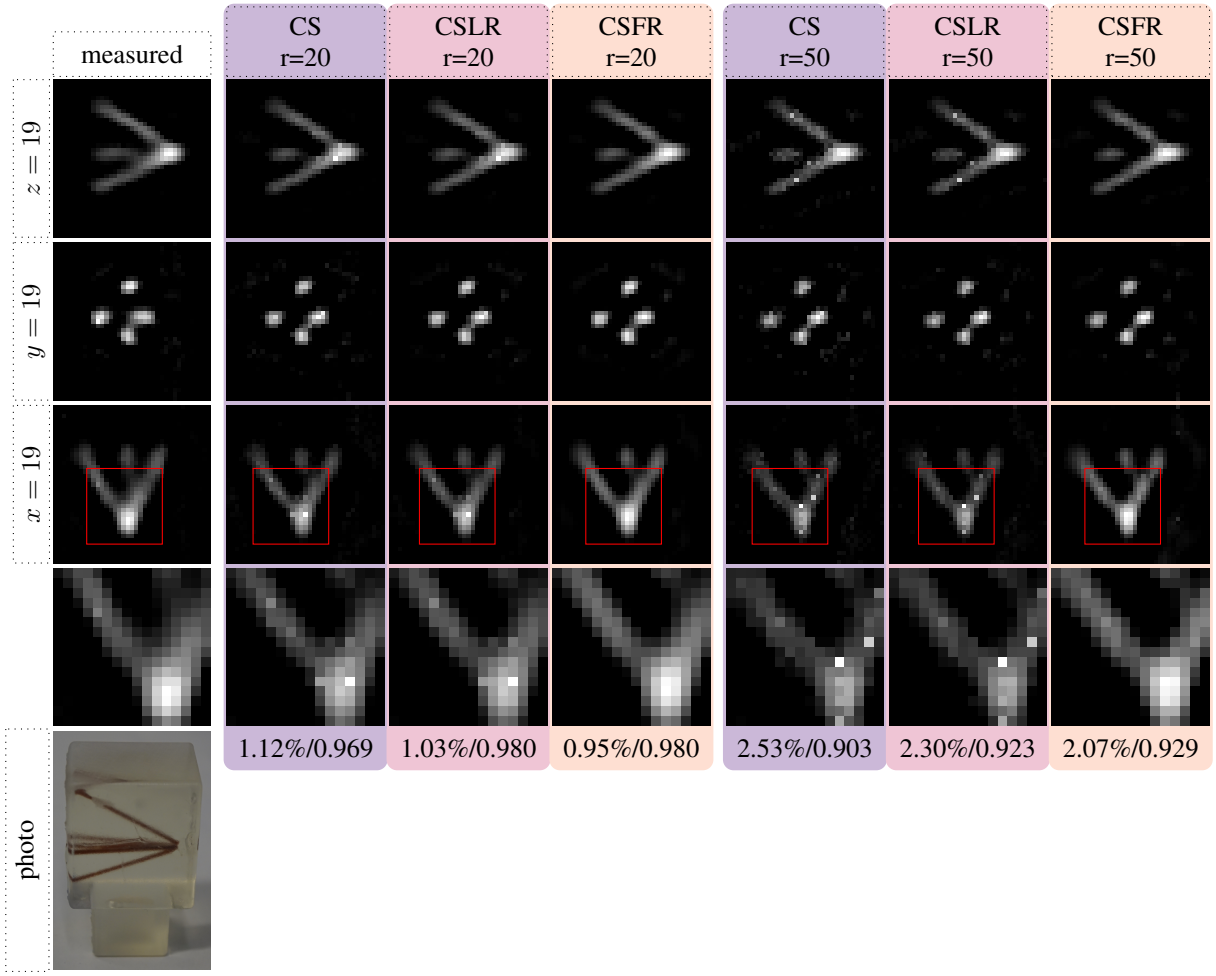


Figure 8.8: Reconstruction of the Open MPI data resolution phantom using system matrices recovered from 20-fold (left) and 50-fold (right) undersampled data. The rows show slices in the xy-, xz- and yz-plane respectively. The numbers below the reconstructed images denote their NRMSD (in %) and their SSIM with respect to the reference reconstruction.

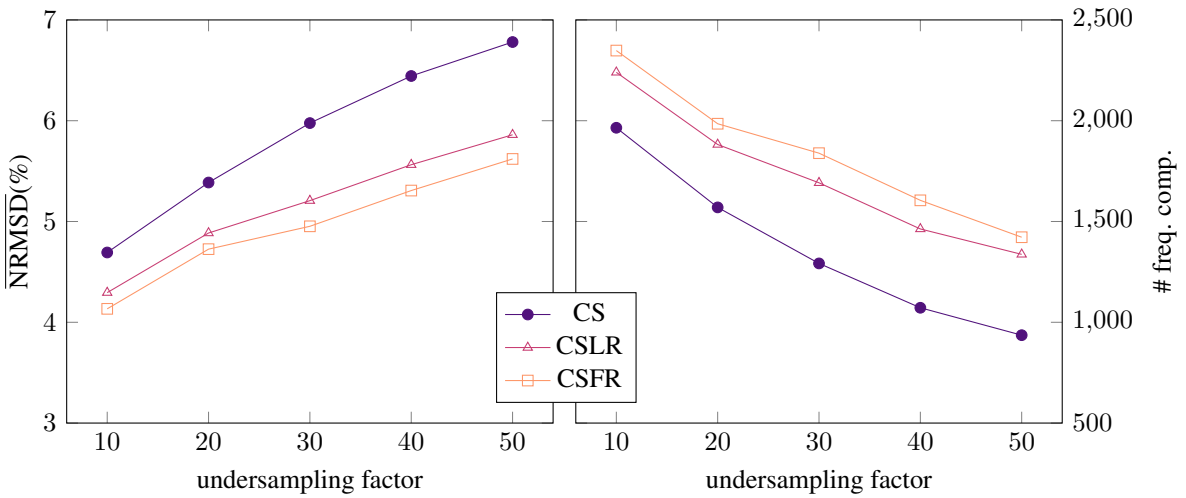


Figure 8.9: Mean NRMSD (left) and number of frequency components with NRMSD smaller than 5% (right) in dependence of the undersampling factor.

and structure of the phantom. Nevertheless, the image reconstruction using the CS-based system matrix displays additional noise induced by errors in the recovered system matrix. For CSLR and CSFR the induced noise and the corresponding values of the NRMSD are reduced, while the SSIM is increased. These results clearly underline the potential of the proposed low-rank methods to provide accurate image reconstruction results while reducing the measurement time for the required system matrix.

A further question to investigate is how the quality of the recovered system matrices is affected when other, potentially larger undersampling factors are used. The results shown in the bottom part of Figure 8.6 show that the general appearance of the patterns is recovered even from 50-fold undersampled data even though the shape of the patterns is slightly altered for all methods. Also the reconstruction results for the resolution phantom show decent quality, notwithstanding that the amount of visible noise is clearly increased compared to the case of 20-fold undersampling. Overall, all of our results show that CSLR and CSFR yield an improved system matrix recovery when compared to CS. An interesting observation is that the improvement in terms of $\overline{\text{NRMSD}}$ remains in the same range for all the undersampling factors considered here. This is illustrated in Figure 8.9, which shows the $\overline{\text{NRMSD}}$ as well as the number of frequency components with NRMSD smaller than 5% in dependence of the undersampling factor. Interestingly, there is a significant improvement even for an undersampling of 10. This is remarkable because the system matrix recovery probem becomes easier to solve when the number of sampling points is increased.

8.7 Discussion

This work is based on the observations that the rows of MPI system matrices can be modeled as low-rank tensors. This form of prior knowledge can be exploited for a variety of tasks such as system matrix

recovery, matrix compression and denoising. Hence, it can help address issues such as the long time needed to measure MPI system matrices as well as noise contamination of the measured data.

Based on the described low-rank property, we developed two methods for the recovery of MPI system matrices and demonstrated that they indeed produce higher quality system matrices than a standard sparsity-based CS recovery. More precisely, modeling the system matrix rows as low-rank tensors leads to an efficient noise suppression in the recovered system matrices. This property is particularly attractive for MPI system matrices because those rows encoding high-resolution information are often corrupted by a significant amount of noise, which renders their recovery more difficult. In line with this observation, the proposed low-rank methods show the largest improvements for frequency components with low SNR.

When comparing CSLR and CSFR, our results show that the additional rank constrained imposed by CSFR yields a stronger denoising than what is achieved by CSLR. This is caused by the more rigorous truncation performed on the HOSVD core tensors. Importantly, this fixed rank constraint induces less bias than the singular value thresholding applied in the solution of the CSLR problem (8.5). Finally, it should be noted that the numerical cost needed for enforcing the fixed rank constraint is negligible in comparison to the cost of solving (8.5).

While providing an increase of quality, the additional regularization terms in CSLR and CSFR also result in a larger number of parameters that need to be set. This is an important aspect to keep in mind when choosing a system matrix recovery scheme. In this work, the regularization parameters were chosen heuristically and those associated to the low-rank regularization were set independently of those associated to the ℓ_1-regularization. A more systematic investigation of the optimization parameters might yield further improvements but was out of the scope of this work. Another interesting approach is to first solve the CS problem (5.26) and apply the fixed rank constraint to this recovery. Although we did not further investigate this approach, it is attractive because it has a lower number of parameters that require tuning. Moreover, solving problem (5.26) instead of (8.5) has a slightly lower numerical cost, which might prove beneficial for use-cases requiring minimal computation times.

An aspect which distinguishes this work from previous works on system matrix recovery is the fact that we investigated the recovery of 3d system matrices, whereas previous works only considered 2d system matrices [28, 29]. For practical applications, this is interesting because the acquisition of 3d system matrices takes a lot longer than that of 2d system matrices. The increased time requirement is caused mainly by the larger size of the 3d reconstruction grid. A positive aspect of the increased spatial dimensionality is that it offers the potential to use larger undersampling factors due to the additional degree of freedom which can be sparsified. In fact, past works on the recovery of 2d system matrices were able to achieve undersampling factors up to 10. In contrast, we were able to recover high quality system matrices using 20-fold undersampled data. Even in the case of 50-fold undersampling we still obtained decent results.

When turning to 3d system matrices, the increased computation time becomes another aspect to keep in mind. On the level of a single frequency components, computation time is typically increased due to

the scaling of operators such as the DCT with the grid size. On the level of the complete system matrix, one should note that 3d system matrices contain a larger number of rows. This is caused by the larger repetition time required for 3d FFP-trajectories to scan the full FOV. For the low-rank methods, the computational costs is increased further due to the fact that three SVDs need to be computed in each iteration of Algorithm 2. In our experiments, the implementation of CSLR and CSFR had a computation time of around 14.2 minutes, whereas the corresponding CS recovery took 4.3 minutes. These computations we were performed using 42 threads on a workstation equiped with two AMD EPYC 7702 CPUS running at 2.0 GHz and a main memory of 1024 GB. To put this into perspective, we note that the measurement of the fully sampled system matrix took 32 hours and 41 minutes. Thus, even the acquisition of the undersampled data would take longer than our computation time for all practical undersampling factors. Based on this, we conclude that the increased computational complexity of CSLR and CSFR does not limit their practical application. Moreover, we note that all methods lend themselves well to parallelization because all frequency components are recovered independently.

From the modeling side, the proposed low-rank methods make use of the Tucker decomposition to quantify the linear dependencies within a given system matrix row. This choice is quite natural in the sense that the Tucker decomposition is the most popularly used technique for these kinds of problems and because it leads to an efficient and intuitive optimization scheme. We note, however, that other decompositions such as the tensor singular value decomposition [157, 167] or the tensor train decomposition [158, 168] have shown promising and even superior results for other low-rank tensor completion tasks. Hence, the application of these more complex decomposition schemes is a promising direction for future work.

We conclude that the low-rank structure of the MPI system matrix rows is a powerful prior with a range of possible applications. Our results show that the proposed low-rank approximation scheme leads to an efficient denoising for MPI system matrices while introducing only negligible amount of bias. Notably, we were able to recover high quality system matrices from 20-fold undersampled data and obtained decent results for undersampling factors up to 50. This reduces measurement time for the corresponding 3d system matrix from the order of a day to the order of a few hours. The proposed methods thus show a great potential to expand the range of feasible MPI applications by reducing the corresponding calibration times. In particular, multi-contrast MPI and multi-patch MPI might benefit from this method as these require the acquisition of a dedicated system matrix for each contrast or patch to be reconstructed.

Part III

Optimized Experiment Design

9 Oracle-Based Experiment Design and its Application in MRI

In this chapter, we introduce a framework for the optimization of the sampling patterns to be used for CS-based data acquisition schemes. The frameworks tailors the sampling patterns to a given set of training data by optimizing the Bayesian Fisher information matrix. After introducing the framework, we demonstrate its application for the generation of MRI sampling patterns. In the following chapter, we further show how the same framework can be used to enhance the CS-based acquisition of MPI system matrices. This work was published in the articles [C1] and [J2].

9.1 Motivation

As we have seen in the previous chapters, the success of image reconstruction methods highly depends on the sampling scheme used during data acquisition. In the field of CS, pseudo-random sampling schemes are a popular choice due to the associated recovery guarantees, which were discussed in subsection 2.7.4. On the other hand, it is well known that the optimal sampling pattern strongly depends on the types of measurements that are performed as well as on the structure of the signals to be recovered. This can be well illustrated using the example of an MRI acquisition. For typical MRI images, a large portion of the measured signal energy is contained in the samples close to the k-space center. This happens because images often contain many structures that are slowly varying in space, such as regions depicting the same type of tissue. These regions are associated with small spatial frequencies and thus contribute strongly to the signal near the k-space center. In contrast, structures with a high spatial frequency are rather sparse, which explains the weaker signal in the outer regions of the k-space. Taking into account that MRI measurements are usually performed in the Fourier domain, this explains the well known fact that variable-density sampling schemes lead to superior results than random schemes with a uniform density.

Since the choice of a suitable sampling scheme can significantly improve the results of experiments, methods tailoring a sampling scheme to the class of signals to be recovered are of great practical value. In MRI this observation has led to the development of dedicated techniques that adapt sampling patterns to a given anatomy at hand [34–36]. An interesting example of such a technique is the OEDIPUS framework, which adapts MRI sampling patterns to the sparsity structure of the underlying images by optimizing

the constrained Cramer-Rao bound [33]. OEDIPUS is inspired by the techniques used in the field of experiment design and thus it is related to a number of more general works tackling the problem of optimal sensor selection [169–171].

Compared to other approaches, one main advantage of OEDIPUS is its clear link to the expected reconstruction error of the image reconstruction method. Moreover, OEDIPUS is designed for an offline computation of sampling patterns, which can then be used for imaging experiments. This makes it very easy to deploy in practice, as the sole requirement is a small training dataset of images with a similar structure as those expected in the imaging experiment. With regard to other works on sampling pattern optimization, we note that [35] only considers support and parallel imaging constraints but does not take into account the sparsity of the images to be recovered. [34] uses tools from Bayesian experiment design and aims to adapt sampling patterns on-the-fly during the actual acquisition. While this approach is interesting, it is also more challenging to implement in practice due to the resulting real-time requirements for the optimization method. An interesting alternative approach is presented in [36]. In this work, the authors propose a bilevel optimization approach, where the sampling pattern is optimized jointly with the regularization parameters of the inverse problem used for image reconstruction. While this approach shows very promising results, its application yields a sampling patterns for one fixed undersampling factor. In contrast, OEDIPUS provides an ordered list of sampling points. This allows one to vary the number of measurements to be performed without having to repeat the computationally expensive sampling pattern optimization.

In this chapter, we present a sampling pattern optimization scheme which can be viewed as a Bayesian extension of the OEDIPUS framework. By optimizing the Bayesian FIM instead of the constrained CRB, we are able to incorporate further information about the signals to be recovered. As a second aspect, we address the long computation times associated with OEDIPUS by a careful optimization of the algorithm used for optimization of the Bayesian FIM.

9.2 Experiment Design

To formalize the problem described in the previous section, we consider the general measurement model (2.13). Additionally, we assume the signal $\mathbf{f}$ to have an S-sparse representation $\boldsymbol{\alpha} = \boldsymbol{\Psi}\mathbf{f}$ with sparsifying transform $\boldsymbol{\Psi} \in \mathbb{C}^{Q\times N}$ as we did in section 2.7. As discussed in chapter 2, recovering the signal $\mathbf{f}$ from the measured data can be viewed as computing a suitable statistical estimator $\hat{\mathbf{f}}$. The sensor selection problem tackled in this chapter is thus concerned with the question of how to choose the set of measurements $\Omega \subseteq \Gamma$ such that the estimation error for $\mathbf{f}$ is minimized [169, 170]. Note that this estimation error depends not only on the measurements performed but also on the estimator used as well as the class of signals to be recovered.

A classical way to assess the achievable accuracy for a given experiment is to compute the CRB. As was discussed in subsection 2.5.1, the latter gives a lower bound for the covariance matrix $\mathrm{cov}(\hat{\mathbf{f}})$ of an unbiased estimator for $\mathbf{f}$

$$\mathrm{cov}(\hat{\mathbf{f}}) \succeq \mathbf{J}_\Omega^{-1}(\hat{\mathbf{f}}) = \left(\mathbf{H}_\Omega^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathbf{y}}^{-1} \mathbf{H}_\Omega\right)^{-1}. \tag{9.1}$$

The matrix $\mathbf{J}_\Omega = \mathbf{H}_\Omega^{\mathsf{H}} \boldsymbol{\Sigma}_{\mathbf{y}}^{-1} \mathbf{H}_\Omega$ is also referred to as the Fisher information matrix. Importantly, (9.1) implies that the diagonal elements of $\mathrm{cov}(\hat{\mathbf{f}})$ can be lower bounded by the corresponding elements of the inverse FIM, i.e. $\mathrm{cov}(\hat{\mathbf{f}})_{ii} \geq \mathbf{J}_\Omega^{-1}(\hat{\mathbf{f}})_{ii}$ where $i \in [1, N]$. Finally, we recapitulate that the right-hand-side of (9.1) corresponds exactly to the covariance matrix of the ML estimator in Eq. (2.21). Thus, the ML estimator actually achieves the lower bound postulated by the CRB.

Based on (9.1), it appears attractive to use $\mathbf{J}_\Omega^{-1}(\hat{\mathbf{f}})$ as the basis for a cost function to optimize the set of measurements to be performed. Thus sensor selection could be performed by searching for a set of measurements, such that the resulting inverse FIM is small in some sense, i.e. by solving an optimization problem of the form

$$\Omega = \underset{\substack{\Omega \subseteq \Gamma \\ |\Omega| = M}}{\operatorname{argmin}} \, F\left(\mathbf{J}_\Omega^{-1}(\hat{\mathbf{f}})\right), \tag{9.2}$$

where M denotes the target number of measurements to be performed. In this formulation, the sense in which $\mathbf{J}_\Omega^{-1}(\hat{\mathbf{f}})$ is minimized is determined by the function $F : \mathbb{C}^{N \times N} \mapsto \mathbb{R}$. One popular choice is the log-det penalty $F_D = \log\det(\cdot)$. According to this formulation, one aims to minimize the size of the confidence ellipsoid associated with $\mathrm{cov}(\hat{\mathbf{f}})$. This notion of minimizing the CRB is also known as D-optimality. Another popular penalty function is $F_A(\cdot) = \mathrm{tr}(\cdot)$, which leads to the concept of A-optimality. Optimizing this function leads to a set of sampling points such that the achievable average variance of $\hat{\mathbf{f}}$ is minimized. This is attractive in so far as it is consistent with the ℓ_2-norm-based data fidelity terms, which are often used in image reconstruction. As a consequence, we only consider the average-variance penalty F_A in this work, even though both approaches would be valid optimization targets.

9.3 Oracle-Based Experiment Design

Based on the CRB, the concepts described in the previous section provide us with a framework to evaluate and optimize sampling schemes for the linear measurement model (2.13). Note, however, that the optimization problem (9.2) only depends on the measurements performed and on the corresponding noise covariance matrix. In particular, this framework does not take into account any information about the sparsity structure of the signal class to be recovered. A second limitation is that the CRB is only valid for unbiased estimators. This is problematic because CS reconstruction schemes are usually biased due to the use of regularization.

To address aforementioned limitations, one can make use of the constrained CRB for sparse parameter vectors, which was introduced by Ben-Haim and Eldar in [172]. A key result of this work is that the CRB for the S-sparse parameter vector $\boldsymbol{\alpha} \in \mathbb{C}^Q$ is equal to the CRB of an estimator bestowed with

oracle knowledge about the locations of the non-zero coefficients of $\boldsymbol{\alpha}$. Thus, let $\boldsymbol{\alpha}_S$ denote the vector containing only the S non-zero coefficients of $\boldsymbol{\alpha}$ and let $\mathbf{U} \in \mathbb{R}^{Q \times S}$ be the matrix that places the coefficients $\boldsymbol{\alpha}_S$ at their respective positions in $\boldsymbol{\alpha}$ (i.e. $\boldsymbol{\alpha} = \mathbf{U}\boldsymbol{\alpha}_S$). In this setting the measured data can be written as

$$\mathbf{y} = \mathbf{H}_\Omega \boldsymbol{\Psi}^\dagger \mathbf{U} \boldsymbol{\alpha}_S + \boldsymbol{\eta}_\Omega. \tag{9.3}$$

The matrix $\mathbf{H}_\Omega \boldsymbol{\Psi}^\dagger \mathbf{U}$ can thus be viewed as a generalized measurement, which generates the measured data $\mathbf{y}$ from the significant coefficients of the sparse representation of $\mathbf{f}$. Based on this setting, the OEDIPUS framework optimizes an oracle-based CRB of the form

$$\operatorname{cov}(\hat{\boldsymbol{\alpha}}_S) \succeq \mathbf{J}_\Omega^{-1}(\hat{\boldsymbol{\alpha}}_S) = \left(\mathbf{U}^\mathsf{H} (\boldsymbol{\Psi}^\dagger)^\mathsf{H} \mathbf{H}_\Omega^\mathsf{H} \boldsymbol{\Sigma}_\mathbf{y}^{-1} \mathbf{H}_\Omega \boldsymbol{\Psi}^\dagger \mathbf{U} \right)^{-1}, \tag{9.4}$$

which can be obtained by inserting the oracle-based measurement matrix matrix $\mathbf{H}_\Omega \boldsymbol{\Psi}^\dagger \mathbf{U}$ into (9.1). Following our previous line of thought, (9.4) can be viewed as the CRB for an estimator which only considers the significant coefficients of $\boldsymbol{\alpha}$.

Here, we further extend the OEDIPUS approach by moving to a Bayesian setting similar to that considered in [170]. In particular, we assume a normal distribution $\mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\boldsymbol{\alpha}})$ with covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{\alpha}}$ as a prior for the coefficients $\boldsymbol{\alpha}_S$. In order to optimize sampling patterns, we propose to optimize the inverse Bayesian FIM

$$\tilde{\mathbf{J}}_\Omega^{-1}(\hat{\boldsymbol{\alpha}}_S) = \left(\boldsymbol{\Sigma}_{\boldsymbol{\alpha}}^{-1} + \mathbf{U}^\mathsf{H} (\boldsymbol{\Psi}^\dagger)^\mathsf{H} (\mathbf{H}_\Omega^\mathsf{H} \boldsymbol{\Sigma}_\mathbf{y}^{-1} \mathbf{H}_\Omega) \boldsymbol{\Psi}^\dagger \mathbf{U} \right)^{-1}. \tag{9.5}$$

This can be interpreted as a Bayesian generalization of the constrained CRB. Moreover, a comparison with (2.27) shows that the inverse Bayesian FIM (9.5) actually corresponds to the covariance matrix for the MAP estimator assuming the oracle-based measurement model and the aforementioned prior on $\boldsymbol{\alpha}_S$. Concerning the choice of a Gaussian prior, we note that this might seem like a pretty strict assumption. However, this type of assumption is quite common in Bayesian statistical modeling. For example, the frequently used ridge regression model (2.26) can be obtained by assuming a linear measurement model with normally distributed measurement noise and a normal prior distribution. Another way to motivate a Gaussian prior is the central limit theorem, which ascertains that the distribution of $\boldsymbol{\alpha}_S$ approaches a Gaussian if $\boldsymbol{\alpha}_S$ is generated by a large number of statistically independent contributions.

Introducing this Bayesian formulation comes with multiple advantages. First of all, the prior covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{\alpha}}$ can be used to incorporate additional knowledge about the signals to be recovered. As an example, let us consider natural images which are sparse in the Wavelet domain. For these it is well known that the Wavelet coefficients associated with a coarse scale typically contain larger values than those associated with fine scales. This form of prior knowledge can easily be modeled by a scale-dependent choice of the variances contained in $\boldsymbol{\Sigma}_{\boldsymbol{\alpha}}$. Secondly, the introduction of a prior guarantees that $\tilde{\mathbf{J}}_\Omega^{-1}(\hat{\boldsymbol{\alpha}}_S)$ is well-defined even in the case of an empty measurement set Ω. This enables us to perform op-

timization of the sampling patterns using both forward and backward-selection approaches. Depending on the target number of measurements, this can lead to a significant reduction of computation times.

Before discussing the solution of the proposed sensor selection problem, we note that the described approach requires prior knowledge about the sparsity structure of the signals to be recovered. For this reason, our method assumes the availability of a number of K_T representative training signals. In practice, each training signal generally has a different sparsity structure, which leads to a different form of the matrices $\mathbf{U}$, $\mathbf{\Sigma}_y$ and $\mathbf{\Sigma}_{\boldsymbol{\alpha}}$ and thus to a different form of $\tilde{\mathbf{J}}_{\Omega}^{-1}$. As a consequence, we aim to minimize the average variance of the oracle-based estimators used for training. Thus, we are searching for a solution of the problem

$$\Omega = \underset{\substack{\Omega \subseteq \Gamma \\ |\Omega| = M}}{\operatorname{argmin}} \sum_{k=1}^{K_T} \operatorname{tr}\left(\tilde{\mathbf{J}}_{\Omega,k}^{-1}(\hat{\boldsymbol{\alpha}}_S)\right), \tag{9.6}$$

where $\tilde{\mathbf{J}}_{\Omega,k}^{-1}$ denotes the inverse of the Bayesian FIM for the k^{th}-training signal. For later reference, we also introduce the notations $\mathbf{U}_k$, $\mathbf{\Sigma}_{\mathbf{y},k}$ and $\mathbf{\Sigma}_{\boldsymbol{\alpha},k}$, which refer to the respective $\mathbf{U}$, $\mathbf{\Sigma}_{\mathbf{y}}$ and $\mathbf{\Sigma}_{\boldsymbol{\alpha}}$ matrices associated with the k^{th} training signal.

9.4 Solving the Sensor Selection Problem

Solving experiment design problems such as (9.6) is a difficult task, because they belong to the class of non-convex integer programming problems. To solve these kinds of combinatorial optimization problems different methods have been proposed, of which some reformulate the problem such that it can be tackled using semi-definite programming [169–171], while others employ greedy algorithms that iteratively add or remove measurements from a given set of measurements [173, 174]. OEDIPUS uses a backwards selection algorithm which starts with a full measurement set $\Omega = \Gamma$ and iteratively removes the least-informative measurements until the target number of measurements is reached [33]. In this work, we instead use a forward selection algorithm, which starts with an empty set and iteratively adds the most informative measurements until the target number of measurements is reached [170]. This is particularly advantageous for large undersampling because the number of measurements to add to Ω becomes significantly smaller than the number of measurements which would need to be removed from Γ until the target number of measurements is reached.

Our procedure for adding one measurement to the set Ω consist of searching for the measurement that leads to the largest decrease of the cost function in problem (9.6) as is summarized in Algorithm 3. The main computational complexity of this procedure arises from the fact that the change of the cost function needs to be computed for each candidate measurement in $\tilde{\Gamma}$. When embedding Algorithm 3 into a full optimization algorithm, the change of the cost function usually needs to be computed $\mathcal{O}(MN)$ times because one needs to perform an exhaustive search over all remaining candidate measurements. For growing problem sizes, this leads to a significant increase in computation time, which can quickly render the algorithm infeasible.

Algorithm 3: Selecting one measurement

Input: set of candidate measurements $\tilde{\Gamma}$
current set of measurements Ω
inverse FIMs $\left\{\tilde{\mathbf{J}}_{\Omega,k}^{-1}\right\}_{k=1}^{K_T}$
matrices $\mathbf{H}_\Gamma$, $\boldsymbol{\Psi}$ and $\{\mathbf{U}_k\}_{k=1}^{K_T}$
covariance matrices $\{\boldsymbol{\Sigma}_{\mathbf{y},k}\}_{k=1}^{K_T}$ and $\{\boldsymbol{\Sigma}_{\boldsymbol{\alpha},k}\}_{k=1}^{K_T}$

1: find the sensor $j \in \tilde{\Gamma}$, which minimizes $\sum_{k=1}^{K_T} \operatorname{tr}\left(\tilde{\mathbf{J}}_{\Omega,k}^{-1}(\hat{\boldsymbol{\alpha}}_S)\right)$
2: add j to Ω and update $\left\{\tilde{\mathbf{J}}_{\Omega,k}^{-1}(\hat{\boldsymbol{\alpha}}_S)\right\}_{k=1}^{K_T}$
3: remove j from $\tilde{\Gamma}$

Output: set of measurements Ω
inverse FIMs $\left\{\tilde{\mathbf{J}}_{\Omega,k}^{-1}\right\}_{k=1}^{K_T}$
remaining candidate measurements $\tilde{\Gamma}$

To alleviate this issue, we note that one can often find multiple measurements, which lead to a similar decrease of the cost function. Based on this assumption, we aim to reduce computation times by temporarily restricting the set of candidate measurements to the most promising ones. Thus, given a temporary measurement set Ω and the corresponding $\left\{\tilde{\mathbf{J}}_{\Omega,k}^{-1}\right\}_{k=1,\dots,K_T}$, we propose the following batch-based procedure for adding more measurements:

1. Compute the change in cost function for all candidates in Γ.
2. Form a batch $\tilde{\Gamma}$ containing the B candidates with the larges decrease of the cost function in (9.6).
3. Iteratively add M_B measurements from the batch $\tilde{\Gamma}$ as described in Algorithm 3.

In a full optimization routine, this procedure needs to be repeated iteratively in order to reach the target number of measurements. Doing so leads to our final algorithm, which is summarized in Algorithm 4. Here, the change of the cost function needs to be computed $\mathcal{O}\left(\frac{MN}{M_B} + MB\right)$ times, which can be a quite significant reduction depending on the chosen values for B and M_B.

When applying the proposed method, the choices of the batch size B and the number M_B of measurements to be added per batch are important parameters to be set. To guide this choice, we note that the regular sequential forward selection algorithm is obtained by setting $M_B = 1$ or $B = |\Omega|$. Note however that one would ideally like to pick M_B as large as possible and B as small as possible in order to minimize computation time. To determine suitable values for these parameters, a good strategy is to perform a number of small test runs, where only a small number of measurements are to be selected. By comparing the results with the sequential forward selection algorithm, on can easily determine a suitable set of parameters, which minimizes computation time without compromising the quality of the measurement set Ω.

Algorithm 4: Greedy algorithm for sensor selection

Input: set of candidate measurements Γ
matrices $\mathbf{H}_\Gamma$, $\mathbf{\Psi}$ and $\{\mathbf{U}_k\}_{k=1}^{K_T}$
covariance matrices $\{\mathbf{\Sigma}_{\mathbf{y},k}\}_{k=1}^{K_T}$ and $\{\mathbf{\Sigma}_{\boldsymbol{\alpha},k}\}_{k=1}^{K_T}$
number of measurements M
batch parameters B and M_B

$\Omega \leftarrow \{\}$
$\left\{\tilde{\mathbf{J}}_{\Omega,k}^{-1}\right\}_{k=1}^{K_T} \leftarrow \{\mathbf{\Sigma}_{\boldsymbol{\alpha},k}\}_{k=1}^{K_T}$
for $i = 1, \ldots, \frac{M}{M_B}$ **do**
  create a batch $\tilde{\Gamma}$ containing the B candidates with the largest decrease of $\sum_{k=1}^{K_T} \operatorname{tr}\left(\tilde{\mathbf{J}}_{\Omega,k}^{-1}(\hat{\boldsymbol{\alpha}}_S)\right)$
  for $k = 1, \ldots, M_B$ **do**
    add one candidate from $\tilde{\Gamma}$ to Ω using Algorithm 3
  end for
end for

Output: set of measurements Ω

9.5 Numerical Implementation

As discussed in the last section, the main numerical bottleneck of Algorithm 3 is the need to compute the change of the cost function for each candidate set. Additionally, the inverse FIMs need to be updated, once the best candidate measurement has been found. In practice, these bottlenecks can quickly render the algorithm infeasible, because all of the aforementioned computations require the inversion of multiple $S \times S$-matrices, thus leading to a computational complexity of $\mathcal{O}\left(K_T \left|\tilde{\Gamma}\right| S^3\right)$.

To perform the discussed computations more efficiently, we assume the noise covariance matrix $\mathbf{\Sigma}_y$ to be a diagonal matrix, i.e. $\mathbf{\Sigma_y} = \operatorname{diag}(\sigma_1, \ldots, \sigma_M)$. As discussed in subsection 4.4.2, this is well justified for MRI measurements, given that noise-whitening has been applied to mitigate channel-cross-talk. Under this assumption, Eq. (9.5) can be written as [33, 170]

$$\tilde{\mathbf{J}}_\Omega^{-1}(\hat{\boldsymbol{\alpha}}_S) = \left(\mathbf{\Sigma}_{\boldsymbol{\alpha}}^{-1} + \sum_{j \in \Omega} \frac{\mathbf{h}_{\boldsymbol{\alpha},j}^* \mathbf{h}_{\boldsymbol{\alpha},j}^\mathsf{T}}{\sigma_j}\right)^{-1}. \tag{9.7}$$

Here $\mathbf{h}_{\boldsymbol{\alpha},j}^\mathsf{T}$ denotes the j^{th} row of the transformed measurement matrix $\mathbf{H}_\Gamma \mathbf{\Psi}^\dagger \mathbf{U}$. Thus, we see that adding a measurement corresponds to a rank-1 update of the matrix $\tilde{\mathbf{J}}_\Omega(\hat{\boldsymbol{\alpha}}_S)$.

To efficiently compute $\tilde{\mathbf{J}}_\Omega^{-1}(\hat{\boldsymbol{\alpha}}_S)$, we follow the approach described in [33, 170, 173] and apply the Sherman-Morrison-Woodbury matrix inversion lemma [175] to (9.7). For this purpose, let $\tilde{\mathbf{J}}_{\Omega+j}(\hat{\boldsymbol{\alpha}}_S)$

denote the FIM obtained when the measurement with index j is added to the experiment. Then, the resulting inverse FIM becomes

$$\tilde{\mathbf{J}}_{\Omega+j}^{-1}(\hat{\boldsymbol{\alpha}}_S) = \tilde{\mathbf{J}}_{\Omega}^{-1}(\hat{\boldsymbol{\alpha}}_S) - \frac{\tilde{\mathbf{J}}_{\Omega}^{-1}(\hat{\boldsymbol{\alpha}}_S)\mathbf{h}_{\boldsymbol{\alpha},j}^{*}\mathbf{h}_{\boldsymbol{\alpha},j}^{\mathsf{T}}\tilde{\mathbf{J}}_{\Omega}^{-1}(\hat{\boldsymbol{\alpha}}_S)}{\sigma_j + \mathbf{h}_{\boldsymbol{\alpha},j}^{\mathsf{T}}\tilde{\mathbf{J}}_{\Omega}^{-1}(\hat{\boldsymbol{\alpha}}_S)\mathbf{h}_{\boldsymbol{\alpha},j}^{*}}. \tag{9.8}$$

Hence, matrix inversion can be circumvented by computing the matrix-vector products on the right hand side of (9.8).

When computing the change in cost function, we are only interested in the trace of $\tilde{\mathbf{J}}_{\Omega+j}^{-1}(\hat{\boldsymbol{\alpha}}_S)$. In this case, the computations can be simplified even further. By making use of the cyclic permutation invariance of the trace one obtains

$$\operatorname{tr}\left(\tilde{\mathbf{J}}_{\Omega+j}^{-1}(\hat{\boldsymbol{\alpha}}_S)\right) = \operatorname{tr}\left(\tilde{\mathbf{J}}_{\Omega}^{-1}(\hat{\boldsymbol{\alpha}}_S)\right) - \frac{\mathbf{h}_{\boldsymbol{\alpha},j}^{\mathsf{T}}\tilde{\mathbf{J}}_{\Omega}^{-2}(\hat{\boldsymbol{\alpha}}_S)\mathbf{h}_{\boldsymbol{\alpha},j}^{*}}{\sigma_j + \mathbf{h}_{\boldsymbol{\alpha},j}^{\mathsf{T}}\tilde{\mathbf{J}}_{\Omega}^{-1}(\hat{\boldsymbol{\alpha}}_S)\mathbf{h}_{\boldsymbol{\alpha},j}^{*}}. \tag{9.9}$$

Thus, the computational complexity of Algorithm 3 is reduced to $\mathcal{O}(K_T|\tilde{\Gamma}|S^2)$.

9.6 Experiments

To test the described pattern optimization method, we simulated two MRI acquisitions based on the complex-valued MRI images accompanying the MEDI toolbox [152, 153]. One of these datasets stems from measurements performed on a Siemens scanner and it contains an image series at 8 echo times $t_{E,i} = 3.6 + 5.9i\,\mathrm{ms}$ $(i = 0, ..., 7)$ and a $256 \times 256 \times 64$ image matrix. The second dataset stems from measurements performed on a GE scanner and it contains an image series at the echo times $t_{E,i} = 2.8 + 5.2i\,\mathrm{ms}$ $(i = 0, ..., 8)$ and a $256 \times 256 \times 74$ image matrix. In our tests, we used the Siemens dataset for the generation of sampling patterns, whereas the GE dataset was used for image reconstruction.

For the generation of sampling patterns, we extracted a single slice along the second dimension of the Siemens dataset and used that to generate independent patterns for all eight echo-images. For this purpose, we used the Daubechies-2 Wavelet transform as sparsifying transfrom. To form the matrix $\mathbf{U}$, which describes the sparsity structure of the images, we selected the top 15% of all Wavelet coefficients in terms of their magnitude. To setup the corresponding prior covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{\alpha}}$, we assume the latter to be diagonal with constant variance $\sigma_{\boldsymbol{\alpha}}$. The remaining variance parameter is estimated directly from the Wavelet coefficients used to setup $\mathbf{U}$. For the noise-variance we assumed a value of $\sigma_y = 1$. To simulate a multi-channel acquisition we furthermore generated a set of four birdcage coil sensitivities, which we used to setup the corresponding multi-channel encoding operator (4.27). Based on these parameters, we used Algorithm 4 to generate sampling patterns with an undersampling factor of 8 for each of the echo images. Here, each batch in Algorithm 4 contained 160 candidate measurements and $M_B = 16$ measurements were added to Ω before updating the batch. For comparison, we also generated a uniform

Poisson disk pattern with a fully sampled 40×30 calibration region in the k-space center and a variable density Poisson disk with a quadratic decay of the sampling density.

Having generated sampling patterns as described, we used the GE dataset to simulate k-space data according to the given sampling schemes. To stay consistent with the image size and number of echoes of the Siemens dataset, we only considered the first 9 echo images of the GE dataset and cropped the images to a size of $256 \times 256 \times 64$. We simulated k-space data by applying the multi-channel encoding operator (4.27) in conjunction with the coil sensitivities described in the last paragraph. After performing undersampling according to the different sampling patterns, multi-contrast image reconstruction was performed.

To obtain a multi-contrast reconstruction, we consider the following image reconstruction problem containing both sparsity-promoting ℓ_1 regularization as well as locally low-rank regularization [58]

$$\underset{\mathbf{m}}{\operatorname{argmin}} \ \|\mathbf{u} - \mathbf{S}\mathbf{m}\|_2^2 + \lambda_1 \|\mathbf{\Psi}\mathbf{m}\|_1 + \lambda_* R_{\mathrm{LLR}}(\mathbf{m}). \tag{9.10}$$

Note that in this case, $\mathbf{m}$ contains the series of all echo images and the sparsity transform $\mathbf{\Psi}$ is applied to all images of this image series. The locally low-rank regularization term aims to exploit spatio-temporal correlations in the image series and it is defined as

$$R_{\mathrm{LLR}}(\mathbf{m}) = \sum_{\mathbf{r}} \|\mathbf{R}_\mathbf{r}\mathbf{m}\|_* \,. \tag{9.11}$$

Here, the linear operator $\mathbf{R}_\mathbf{r}$ extracts a $p_x \times p_y$-sized patch centered at $\mathbf{r}$ from $\mathbf{m}$ and reshapes it into a matrix of size $p_x p_y \times N_E$ with N_E denoting the number of echo images. For typical MRI image series this local Casorati-matrix is expected to have a low rank and thus R_{LLR} uses the nuclear norm, defined in (8.6), to enforce this behavior. To motivate this low-rank structure, we recapitulate that the temporal evolution of the signal associated with a given voxel is determined by the relaxation times of the underlying tissue. This leads to a small rank of the local Casorati matrix whenever only a few types of tissue are present in a given patch.

To solve the image reconstruction problem we used the ADMM (9.10) [176]. More details on this method can be found in subsection A.2.2 where we derive the ADMM iterations (A.27) using the concept of Bregman iteration. For the given task, the regularization parameters were heuristically chosen as $\lambda_1 = 0.01$ and $\lambda_* = 0.2$ and the locally low-rank regularization was performed using batches of size 8×4. Finally, the ρ-parameters of the ADMM were both set to a value of 0.1 and the algorithm was ran for 100 iterations.

9.7 Results

The patterns obtained using Algorithm 4 are shown in Figure 9.1 for the first and the fifth echo image. As expected, the patterns display a varying sampling density due to the high prevalence of slowly varying

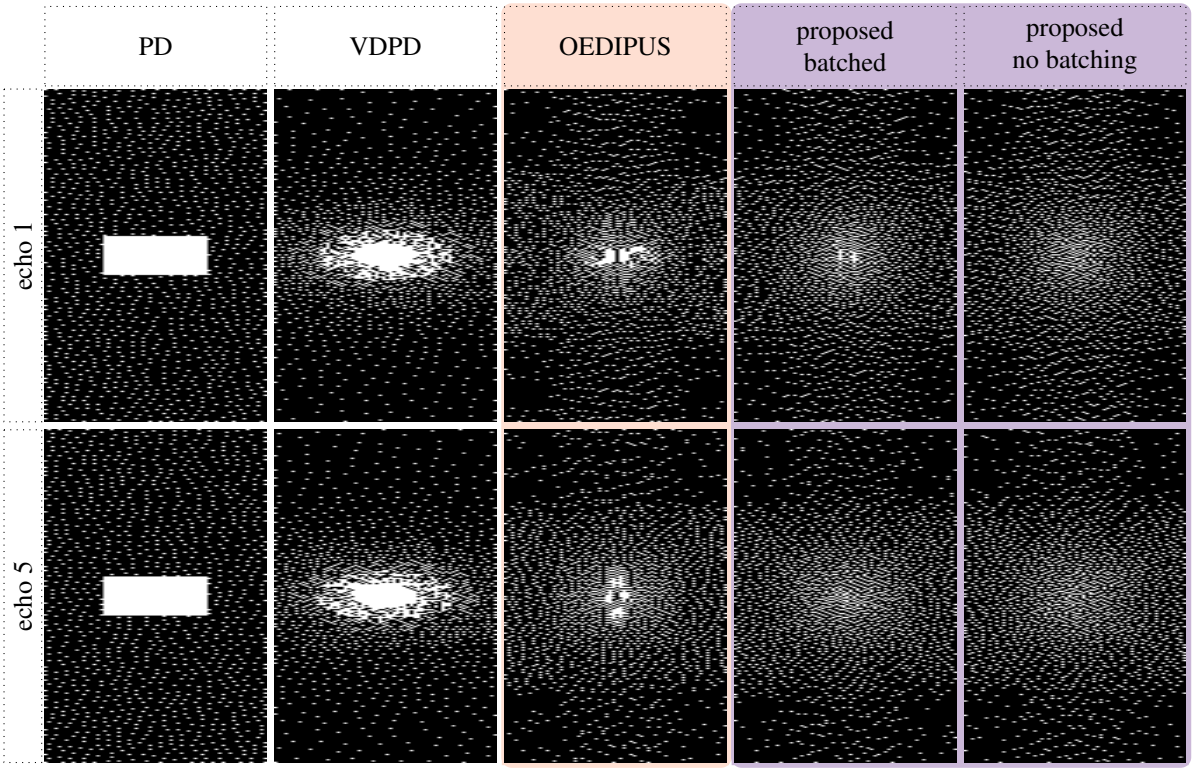


Figure 9.1: Sampling patterns generated by using Poisson disk sampling, variable density Poisson disk sampling, OEDIPUS, the proposed method including batching and the proposed method without batching (from left to right). The patterns are shown for the first echo (top row) and the fifth echo (bottom row).

structures in the training images. When comparing the results to those obtained using the OEDIPUS method, one observes a similar spatial dependence of the sampling density. Moreover, one can see that the OEDIPUS patterns show a slightly higher concentration of measurements in the k-space center. This most likely is a result of the additional prior used in our Bayesian method.

An interesting aspect of the obtained patterns is that the sampling density also changes for the different echo images. These changes are driven by multiple mechanisms. On the one hand, the image contrast varies across echo times due to the different T_2^*-relaxation times of the various tissues. On the other hand, B_0-inhomogeneity leads to an accrual of image phase as the echo time is increased. Both of these mechanisms have an influence on the sparsity structure of the echo images and thus affect the sampling density of the generated sampling patterns.

When comparing the results of the proposed method with and without batching one finds that the obtained sampling pattern show a very similar, albeit not identical, structure. These slight differences can be expected due to the non-convex nature of the sensor selection problem. Nevertheless, our results indicate that the overall quality of our patterns is not compromised by the proposed batching approach for the chosen parameters.

A significant difference between our method and the original OEDIPUS method are the computation times required for generation of the sampling patterns. Using two threads on a workstation equipped with two AMD EPYC 7702 CPUs, the generation of one sampling pattern with the proposed method (including batching) took $\sim 2\,\mathrm{h}$. In contrast, generation of one pattern using the proposed method without batching took $\sim 19\,\mathrm{h}$ and generation of one pattern using OEDIPUS took $\sim 3.5\,\mathrm{d}$. For the multi-echo setting considered here, 8 such patterns need to be generated leading to a corresponding increase in computation time. In this context, the improved computation time of the proposed method appears even more significant. These results are also visible in the error plots in the bottom two rows of Figure 9.2 and the corresponding values of the NRMSD, which are superimposed onto the reconstructed images.

In our image reconstruction experiments, the optimized sampling patterns lead to a clear improvement of the reconstructed images compared to heuristic pseudo-random sampling patterns. In Figure 9.2 this is illustrated for one representative slice of the GE dataset. As one can see, the images obtained using Poisson disk sampling display clearly visible blurring artifacts. When using variable density Poisson disk sampling, these artifacts are strongly reduced and the overall image quality is improved. Nevertheless, image quality is increased even further when using the sampling patterns obtained by Algorithm 4. When comparing the patterns obtained with and without batching, one finds that the reconstructed images are visually almost indistinguishable. Similarly the resulting values of the NRMSD differ only in the order of 0.1%, which is small when compared to the differences between the Poisson disk based and the optimization based sampling patterns. To obtain a more comprehensive overview of the reconstruction quality of the different patterns, Figure 9.3 shows the $\overline{\mathrm{NRMSD}}$ of the reconstructed images as a function of slice position as well as the different contrasts. When comparing the different sampling schemes these results also illustrate the improvements obtainable by tailoring sampling schemes to the anatomy at hand. Another interesting observation is that the optimized sampling patterns yield a similar improvement in NRMSD over almost all slices and echo images considered in this experiment. This shows that the optimized sampling patterns also perform well on other images with a different sparsity structure than the image used for the pattern generation.

9.8 Discussion

This work is based on the observation that the performance of a given MRI sampling pattern depends strongly on the structure of the object being imaged. Thus, imaging of a given anatomy could be improved by using sampling patterns that are optimized towards the imaging of that anatomy. In the context of CS, the sparsity structure of the images plays a central role as this structure determines to what extend data can be undersampled while still allowing for a high-quality image reconstruction. Based on these observations, the proposed method aims to tailor sampling schemes to the expected sparsity structure of the objects to be imaged. To quantify the achievable reconstruction quality, our method uses the Bayesian FIM which can be viewed as a Bayesian extension of the Cramer-Rao bound and which provides an esti-

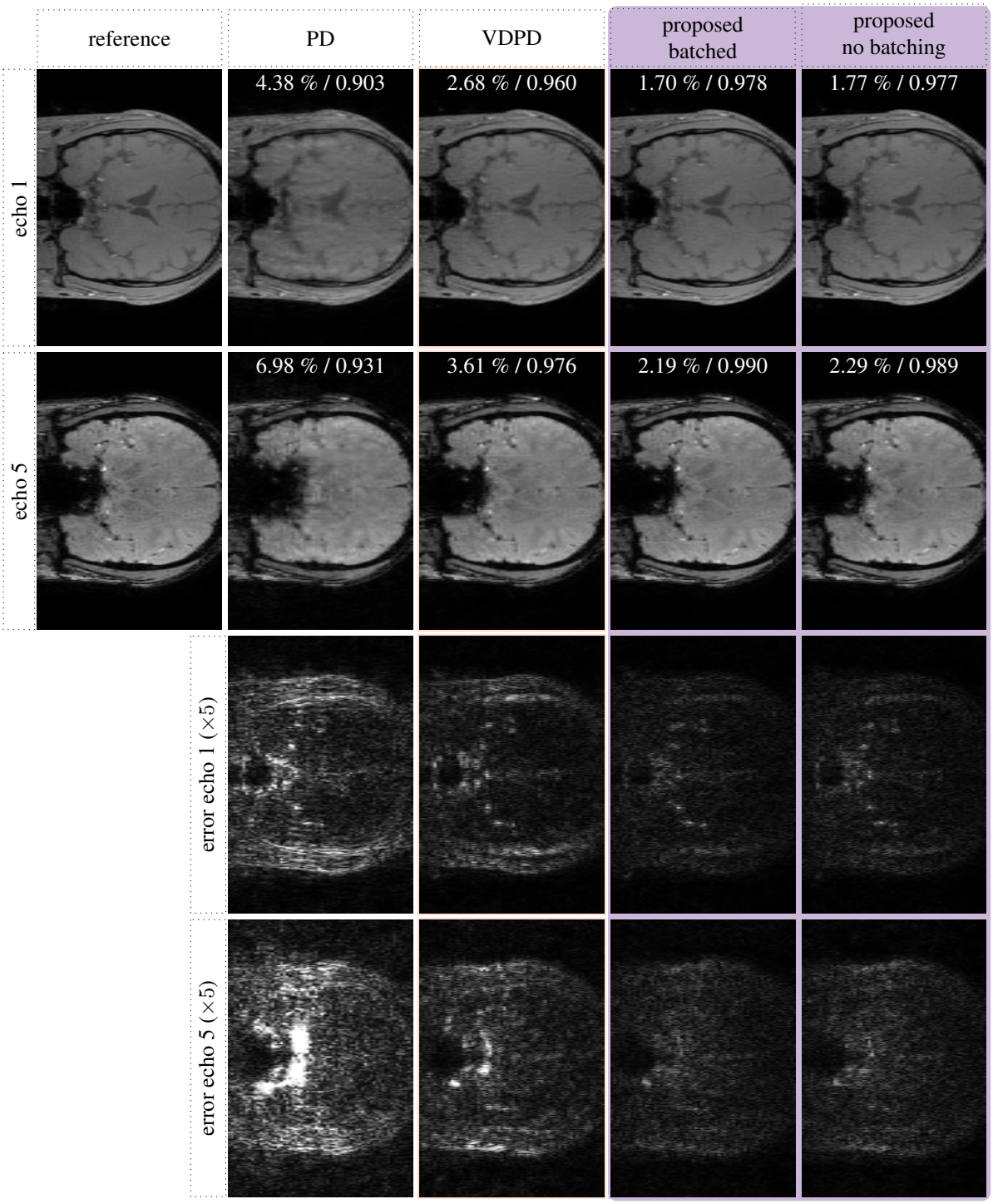


Figure 9.2: Slice of the images reconstructed using the generated sampling patterns. Images are shown for the first echo (top row) and the fifth echo (second row). The superimposed numbers give the NRMSD and the SSIM of the corresponding image with respect to the fully sampled reference image. The third and fourth row display the difference plots compared to the fully sampled reference.

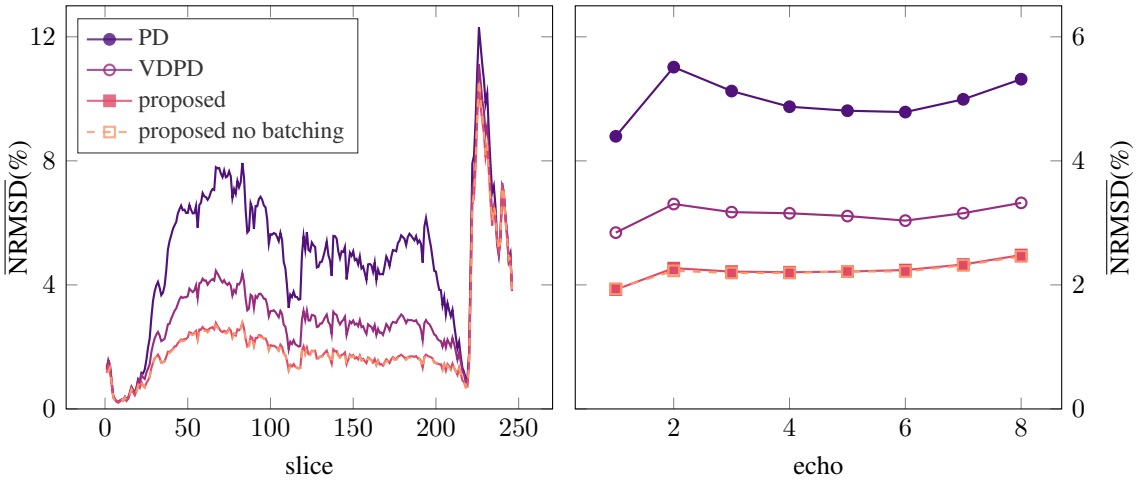


Figure 9.3: $\overline{\text{NRMSD}}$ of the reconstructed images as a function of the slice position (left) and the contrasts (right).

mate for the achievable covariance matrix when performing image reconstruction with a given sampling scheme. Note that the described method is very generic. Thus, its application is not restricted to MRI but it can also be used in other settings which rely on a linear measurement model. As an example, chapter 10 discusses how it can be used to enhance the sampling and recovery of MPI system matrices.

The described method is inspired by the OEDIPUS framework. However, the main difference is the usage of Bayesian FIM instead of the constrained CRB. This move from a frequentist setting to a Bayesian setting comes with multiple advantages. Most importantly, the Gaussian prior allows us to use sequential forward selection algorithms, which stand in contrast to the sequential backward selection algorithm used by OEDIPUS. This modification can significantly reduce computation times especially for the case of highly undersampled acquisitions. A second advantage is that the prior covariance matrix can be used to inject further knowledge about the images to be recovered into our model. While this second aspect does not play a major role in the example considered here, we note that it will play an important role when applying Algorithm 4 to the sampling of MPI system matrices, as we do in chapter 10.

In our example, the forward selection algorithm in combination with the described batching scheme allowed us to reduce computation times by a factor of ~ 40 without compromising the quality of the obtained sampling patterns. A consequence of this acceleration is that it becomes feasible to generate optimized sampling patterns for a wider range of applications than what is possible with OEDIPUS. To put this into perspective, we note that we envision the proposed method to be used for an offline computation of sampling patterns, which can then be used for future imaging applications. With this in mind, computation time is not as important as it is for image reconstruction. Nevertheless, computation time still plays an important role for the adoption of sampling pattern optimization methods. To illustrate this, we note that in clinical applications, sampling patterns would need to be generated for a large range of anatomies and imaging contrasts. This is only possible if the used method does not require excessive amounts of computation time.

With regard to our optimization method, we note that (9.6) is a non-convex, combinatorial optimization problem. Thus, we cannot hope to find a global minimizer without performing an exhaustive search, which is infeasible for the high-dimensional problems considered here. Taking this into account, we make no claims concerning the optimality of the sampling patterns obtained using our method. Nevertheless, our results indicate a clear improvement of our optimization-based approach compared to heuristic sampling patterns.

10 Optimized Sampling Strategies for MPI System Matrices

In this chapter, we investigate how to optimally choose the sampling points for MPI system matrices. To address this question we use the experiment design method described in chapter 9 and apply it to previously measured system matrices, which we use as training data. We also show how the prior covariance matrix can be used to model the characteristic spatial distribution of signal energy in the MPI system matrix rows. This work was published in the article [J2].

10.1 Motivation

As we have seen in the previous chapter, CS-based image reconstruction can be greatly improved by using sampling patterns that are tailored to the structure of the signals to be recovered. In the field of MRI, it has long been known that variable-density sampling patterns could facilitate image reconstruction due to the large signal intensities in the low frequency regions of the k-space. This led to the development of more systematic approaches for the determination of suitable sampling schemes [34, 36, 87].

In contrast, previous methods on the recovery of MPI system matrices have all focused on pseudo-random sampling schemes, due to the associated recovery guarantees which are available in the literature. Nevertheless, it can be observed that the energy contained in the MPI system matrix shows a characteristic spatial structure with large values in the center of the FOV and decaying values towards the FOV edges. This immediately leads to the question whether MPI system matrix recovery would also benefit from optimized sampling patterns that take into account this spatial structure.

In this chapter, we address this question by tailoring sampling patterns to the structure of previously measured system matrices. To approach this task, we make use of the sensor selection scheme developed in chapter 9. Moreover, we use the prior covariance matrix to model the spatial distribution of signal energy in the given system matrices. In addition to the insights about system matrix recovery, this experiment also constitutes a further validation of the sensor selection scheme proposed in Algorithm 4.

10.2 Spatial Structure of the MPI System Matrices

As was discussed in chapter 5, the rows of the MPI system matrix show a distinct oscillatory structure, which is illustrated in Figure 5.2 and which can be roughly modeled as a convolved tensor product of Chebyshev polynomials of the second kind [118]. This oscillatory structure is interesting as it determines the sparsity structure of the system matrix rows. If a sparsifying transform, such as the DCT, is applied to a given frequency component, the result will exhibit a peak at the frequency that matches the corresponding oscillation frequency of the given pattern. Moreover, it can be expected that the other non-zero coefficients will be contained in a bounded region around the central frequency [161]. This structure appears very promising as it can be exploited for the generation of optimized sampling patterns. It should be noted, however, that the measurement process of the MPI system matrix is designed such that all frequency components are acquired jointly with the same sampling pattern. Thus, the final sampling pattern should not be tailored to a single sparsity pattern. Instead it should allow an efficient recovery of all system matrix rows with their respective sparsity structure.

A more generic structure that is not specific to a given system matrix row is the spatial distribution of the signal energy of the respective frequency components. Interestingly, this distribution typically has non-uniform structure with larger values in the center of the FOV and small values as one moves outwards away from the DF-FOV. To understand this behavior, consider a sample of particles in the center of the FOV. In this case, the field at the position of the sample often changes sign according to its sinusoidal nature. These dynamics lead to a strong magnetization dynamics and thus a correspondingly strong particle signal is induced in the measured signal. In contrast, a sample outside the DF-FOV experiences a field, which never changes sign due to the large magnitude of the selection field. In turn, the SPIOs are always close to saturation and the induced particle signal is correspondingly small. This behavior can also be seen in Figure 5.2, where the shown frequency components mainly take on small values near the borders of the FOV. For the corresponding Chebyshev-approximations in Figure 5.2, this structure is even more visible. The reason is that the approximation in terms of Chebyshev polynomials effectively considers the limit of large particles with a step like magnetization curve. As a result, particles outside the DF-FOV are always in saturation and thus do not induce a measurable MPI signal.

10.3 Optimizing Sampling Points for the MPI System Matrix

To generate optimized sampling patterns for MPI system matrices, we propose to use an existing fully sampled system matrix as training data. After optimization, the adapted sampling pattern can be used for the measurement of other system matrices. This setting is realistic because one regularly ends up having to measure new system matrices in MPI. This happens for example when performing experiments under different physical conditions (e.g. different temperature or viscosity), when changing acquisition parameters such as the drive-field amplitude or when maintenance work is performed at the MPI scanner.

As all system matrix rows are obtained using the same sampling pattern, the training data should contain a representative set of all frequency components of the system matrix. At this point it is important to note that using all frequency components quickly becomes impractical due to their large number, especially for 3d system matrices. Hence, care needs to be taken to ensure that the frequency components used for training indeed cover the range of sparsity patterns present in typical system matrices. If this is not the case, there is an increased risk that the optimized sampling patterns do not perform well when used for the recovery of other system matrices. A simple way to ensure a sufficient diversity in the training data is to use a number of pseudo-randomly chosen frequency components. We propose to make this choice in combination with an SNR threshold. The latter ensures that the chosen frequency components contain a sufficient amount of information to inform our experiment design method.

In addition to the sparsity structure of the system matrix, we also aim to take into account the spatially varying distribution of signal energy in the system matrix rows. This behavior can be modeled by a suitable choice for the prior covariance matrix $\mathbf{\Sigma}_{\boldsymbol{\alpha}}$. To achieve this, we use a diagonal covariance matrix $\mathbf{\Sigma}_{\mathbf{s}}$ with spatially dependent variances. For each voxel, the latter is chosen proportional to the variance of the training data at this position. Thus, we use a covariance matrix of the form

$$\mathbf{\Sigma}_{\mathbf{s}} = \mu \operatorname{diag}\left(\operatorname{var}(\{\mathbf{s}_{k,:}\}_{k=1,\dots,K_T})\right). \tag{10.1}$$

Here the set $\{\mathbf{s}_{k,:}\}_{k=1,\dots,K_T}$ contains all k_T frequency components that are used for training and the variances are computed component-wise over the training set. In order to apply this covariance matrix with the Bayesian FIM in (9.6), $\mathbf{\Sigma}_{\mathbf{s}}$ needs to be transformed into the space spanned by the sparse coefficients $\boldsymbol{\alpha}_S$. Thus, our prior covariance matrix takes the form

$$\mathbf{\Sigma}_{\boldsymbol{\alpha}} = \mathbf{U}^{\mathsf{H}} \left(\mathbf{\Psi}^{\dagger}\right)^{\mathsf{H}} \mathbf{\Sigma}_{\mathbf{s}} \mathbf{\Psi}^{\dagger} \mathbf{U}. \tag{10.2}$$

Having introduced our choice for the prior covariance matrix, optimization of sampling patterns can be performed by solving the sensor selection problem (9.6). By putting everything together and using Algorithm 4 for the solution of (9.6), we obtain our final method which is summarized in Algorithm 5.

10.4 Experiments

10.4.1 Datasets

To evaluate the proposed method we employed both 2d and 3d datasets. In both cases, two system matrices measured with different tracer materials where used. Respectively, one of the system matrices was used to generate optimized sampling patterns according to Algorithm 5. For testing purposes, the other system matrix was undersampled according to the obtained patterns and system matrix recovery was performed using the undersampled data.

Algorithm 5: Optimizing sampling patterns for MPI system matrices

Input: system matrix $\mathbf{S}$
fully sampled measurement matrix $\mathbf{H}_\Gamma$
sparsifying transform $\mathbf{\Psi}$
number of measurements M
batch parameters B and M_B

1: $\Gamma \leftarrow \{1, 2, \ldots, N\}$
2: $\tilde{\mathbf{S}} \leftarrow$ representative subset of K_T rows from $\mathbf{S}$
3: $\{\mathbf{U}_k\}_{k=1}^{K_T} \leftarrow$ sparsity patterns of the rows in $\tilde{\mathbf{S}}$
4: $\mathbf{\Sigma_s} \leftarrow$ signal variances for each voxel
5: $\{\mathbf{\Sigma}_{\boldsymbol{\alpha},k}\}_{k=1}^{K_T} \leftarrow \mathbf{U}_k^{\mathsf{H}} \left(\mathbf{\Psi}^\dagger\right)^{\mathsf{H}} \mathbf{\Sigma_s} \mathbf{\Psi}^\dagger \mathbf{U}_k$
6: $\{\mathbf{\Sigma}_{\mathbf{y},k}\}_{k=1}^{K_T} \leftarrow$ noise variances of $\tilde{\mathbf{S}}$
7: $\Omega \leftarrow$ choose M measurements using Algorithm 4

Output: set of measurements Ω

As in chapter 8, the system matrices used for this experiment were measured with a preclinical Bruker 15/20FF MPI scanner (Bruker Corp., Ettlingen, Germany). The two 2d system matrices were obtained using the MP tracer Perimag (micromod Partikeltechnologie GmbH, Rostock, Germany). In one case, liquid particles were used for the measurements. In the other case the particles were immobilized using dental cement prior to the measurements. This immobilization effectively suppresses the Brownian relaxation pathway of the particles and thus significantly alters the structure of the corresponding system matrix. Both system matrices were acquired on a FOV of $(30 \times 30 \times 1)\,\mathrm{mm}^3$ with a grid size of $30 \times 30 \times 1$. The drive-field amplitudes were 12, 12, and 0 mT/μ_0 and the selection field gradients were set to −1, −1, and 2 T/(μ_0 m). In the 2d experiment, generation of the sampling patterns was performed using the system matrix measured with liquid particles. Correspondingly, testing was performed using the system matrix measured with immobilized particles.

Additionally, we used two 3d system matrices from the Open MPI Data repository (calibration datasets 6 and 7) [113]. One of them was acquired used the tracer Perimag, whereas Synomag-D (micromod Partikeltechnologie GmbH, Rostock, Germany) particles were used for the other system matrix. Both system matrices were acquired on a FOV of $(37 \times 37 \times 18.5)\,\mathrm{mm}^3$ with a grid size of $37 \times 37 \times 37$. The drive-field amplitudes were 12, 12, and 12 mT/μ_0 and the selection field gradients were set to −1, −1, and 2 T/(μ_0 m). In the 3d experiment, we used the SM measured with Synomag-D particles for the generation of sampling patterns. For testing, we used the system matrix measured with Perimag particles.

10.4.2 Sampling Patterns

For the generation of the optimized sampling patterns, we applied an SNR threshold of 3 and made a selection of 30 random frequency components for the 2d case and 15 random frequency components for

the 3d case. As in chapter 8 we used the DCT-II as sparsifying transform. With this at hand, we generated a sparse approximation by applying a threshold to the transform coefficients. For the 2d (3d) case we kept the 10 % (2%) largest coefficients and used their locations to form the corresponding $\mathbf{U}$-matrices.

Before fixing the covariance matrices of the measurement noise and of the prior distribution, we normalized all frequency components by their ℓ_2-norm. The variances for the prior distributions were then calculated according to (10.2), where we set the proportionality factor to $\mu = 3$ for both the 2d and the 3d case.

To fix the noise variances, we assumed a constant value $\sigma(\mathbf{s}_k)$ for all sampling points. For its estimation we further assumed that the signal stays approximately constant in the corners of the FOV. This is a reasonable assumption due to the size of the overscan used in these measurements. The latter ensures that particles in the corners of the FOV remain far from the FFP during the measurement. Thus, their signal contribution is close to zero and slowly varying in space. Based on this assumption, we consider the outermost 3×3 voxels of each corner for the 2d case and the outermost $3 \times 3 \times 3$ voxels for the 3d case. Now let $\mathbf{s}^{\text{bg}}_{k,:}$ contain the values of the given frequency component $\mathbf{s}_k$ in these corner voxels. Then our estimate can be written as

$$r(\mathbf{s}_k) = \frac{\left\| \mathbf{s}^{\text{bg}}_{k,:} - \bar{\mathbf{s}}^{\text{bg}}_{k,:} \right\|_2^2}{\left\| \mathbf{s}^{\text{bg}}_{k,:} \right\|_0 - 1}. \tag{10.3}$$

Here each entry in $\bar{\mathbf{s}}^{\text{bg}}_{k,:}$ corresponds to the mean of the values in the corresponding corners of the FOV.

With the training data at hand, we used Algorithm 5, to obtain adapted sampling patterns. For the 2d case, this was done using a batch size of $B = 180$ and $M_B = 4$. For reference, we also generated a 2d pattern without batching (i.e setting $B = N$ and $M_B = 1$). For the 3d case, we used a batch size of $B = 5000$ and $M_B = 8$.

10.4.3 System Matrix Recovery

To test the obtained patterns we generated correspondingly undersampled data from the 2d system matrix with immobilized Perimag and the 3d system matrix with liquid Perimag. For the 2d experiment, we created datasets with undersampling factor 2, 4, 6, 8 and 10. For the 3d case we consider an undersampling factor of 20. For comparison we also generated Poisson disk (PD) sampling patterns with the same undersampling factors. With the undersampled data at hand, we used the Split Bregman method to solve the CS problem (5.26) for all frequency components with an SNR > 3.

For the 2d experiment, matrix recovery was performed with a selection of regularization parameters

$$\lambda_j = \left(\frac{2}{5}\right)^j, \quad \text{for} \quad j = 3, \dots, 8.$$

For the 3d case the regularization parameter was manually chosen such that the mean NRMSD was minimized. This resulted in a value of $\lambda = 0.005$ for the PD pattern and a value of $\lambda = 0.002$ for the optimized sampling pattern. In all cases the solver was ran for 50 inner and 10 outer iterations.

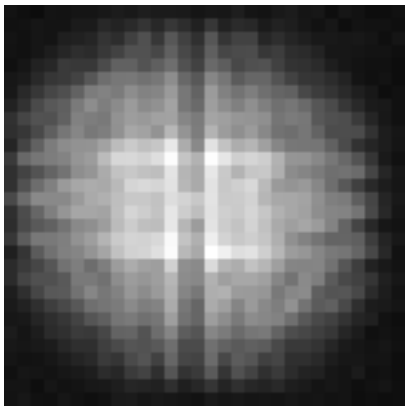


Figure 10.1: Diagonal of the spatial prior covariance matrix $\mathbf{\Sigma_s}$ obtained from the 30 frequency components used for generating the 2d sampling patterns.

10.4.4 Experimental Evaluation

Our evaluation of the recovered system matrices uses the same principles as we did in chapter 8. Hence, we computed the NRMSD and the SSIM for the recovered frequency components in order to quantify their similarity to the fully sampled reference. As an aggregate measure we compute the mean NRMSD and the mean SSIM. Finally, we used the recovered system matrix to perform image reconstruction of the resolution phantom and the shape phantom dataset, which are contained in the Open MPI Data repository. These image reconstructions serve as an indicator for the kind of image quality that can be obtained when the recovered system matrices are used in actual applications. Image reconstruction was performed using the ℓ_2-regularized Kaczmarz algorithm with 10 iterations and a relative regularization parameter of 0.0025.

10.5 Results

10.5.1 2d System Matrices

An important parameter of the proposed method is the prior covariance matrix $\mathbf{\Sigma}_\alpha$ as it has a significant influence on the sampling density of the resulting sampling patterns. In Figure 10.1, we show its spatial representation $\mathbf{\Sigma_s}$ obtained using (10.1). As expected, the signal intensity is largest in the center of the FOV and decays towards its borders. Hence, one can expect a similar spatial dependence of the sampling density of the optimized patterns.

For illustration, we plot the obtained sampling patterns for undersampling factors 2, 6 and 10 in the middle and bottom row of Figure 10.2. Especially for the cases of 6- and 10-fold undersampling, one observes that the sampling density is increased in the center of the FOV, whereas it is decreased outside the DF-FOV. This observation can be explained by the spatially dependent variance of $\mathbf{\Sigma_s}$ employed in our model. Interestingly, a comparably large number of sampling points are placed on the borders of the

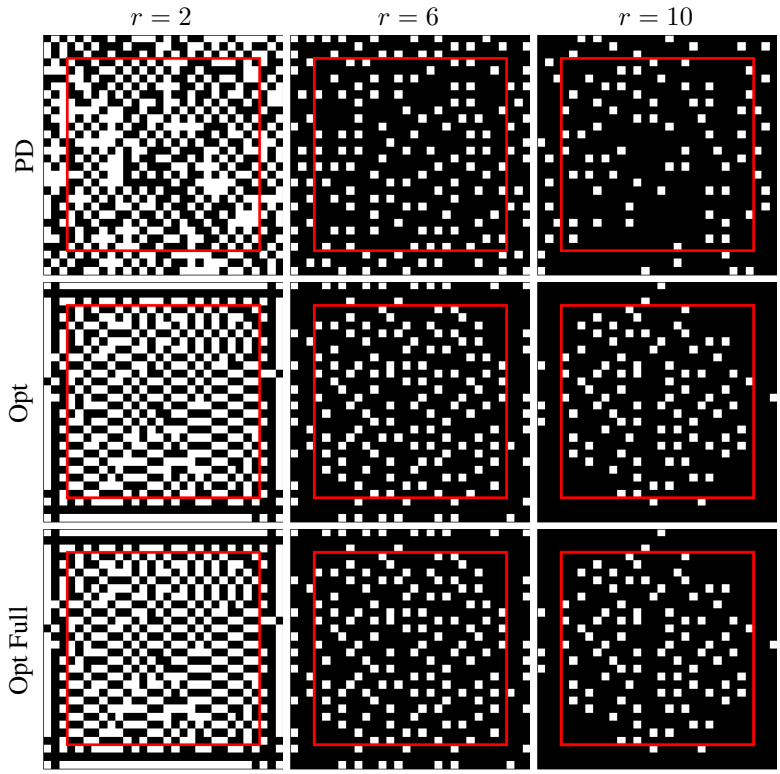


Figure 10.2: 2d sampling patterns generated using Poisson disk sampling (top row), the proposed method (middle row) and the proposed method without batching (bottom row). The patterns are shown for undersampling factors of 2,6 and 10. For reference, the borders of the DF-FOV are indicated by red lines.

FOV. These could be interpreted as control points, which help to accurately recover the missing information in the more sparsely sampled outside regions of the FOV. Besides that, no additional structures or clustering can be observed. Most likely, this reflects the diversity of frequency components used for training. As a consequence, we expect the obtained sampling patterns to be sufficiently robust to recover other unseen frequency components with a similar spatial distribution of intensity.

We also note that the sampling patterns obtained with batching look almost identical to the ones obtained without batching. This indicates that the batching approach can indeed be used to accelerate the solution of the sensor selection problem without significantly worsening the quality of the solution.

The spatially dependent sampling density has a direct influence on the recovery error of the frequency components. Due to the larger number of samples in the center of the FOV, one expects smaller recovery errors in that region. In contrast, the error might be slightly increased outside the DF-FOV. When considering the system matrices recovered in our experiments, the reduction of the reconstruction error inside the DF-FOV can clearly be seen. As an illustration, we show a selection of recovered frequency components in Figure 10.3, along with their recovery error. The reduction of the error in the center of the FOV can be seen for the frequency components with $k = 59$ and $k = 62$. The situation is different for

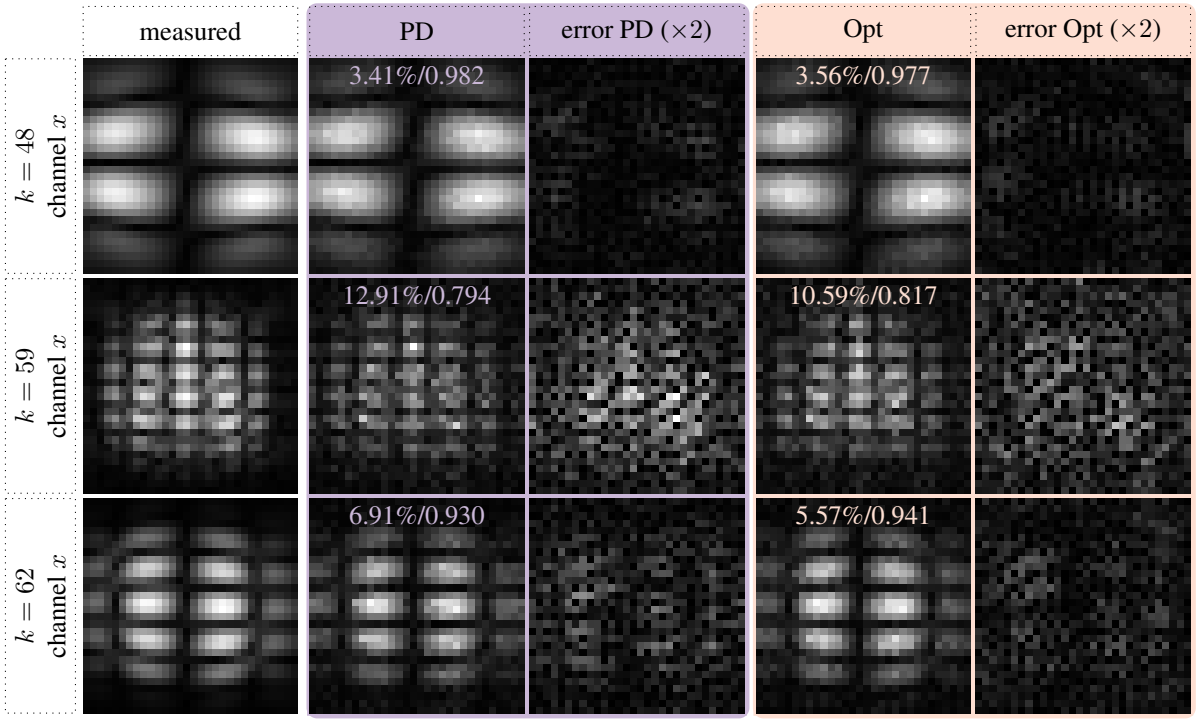


Figure 10.3: Recovered frequency components from 6-fold undersampled data. The second and third column show the patterns obtained from PD sampling and the respective error plots. The fourth and fifth column show the corresponding quantities for patterns obtained using Algorithm 5. The NRMSD (in %) and the SSIM with respect o the fully sampled reference are superimposed onto the plots. All plots in a given row are scaled to the same colorbar. The error plots where scaled by a factor of 2 to increase their contrast within the given colorbar.

the frequency component with $k = 48$. In this case the PD sampling shows a slightly better result. This is probably caused by the extended structure of the frequency component, which benefits from a more uniform sampling.

Since the signal intensity tends to be largest in the DF-FOV, the non-uniform sampling leads to an overall decrease in recovery error. For the plotted frequency components, this is reflected in the NRMSD values superimposed onto the corresponding plots. Additionally, we plot the NRMSD of the recovered frequency components in dependence of their SNR in the Box-Whiskers plot in Figure 10.4. This plot shows an overall reduction in NRMSD irrespective of the SNR of the frequency components. Interestingly, a corresponding reduction in $\overline{\text{NRMSD}}$ can be observed for all regularization parameters considered, as can be seen in the upper panel of Figure 10.5. Finally, it should be noted that the effect of the non-uniform sampling becomes more pronounced with increasing undersampling factors. This is understandable because the chance that a random sampling pattern will miss a measurement with high intensity (i.e. with high information content) increases the fewer the number of measurements. This

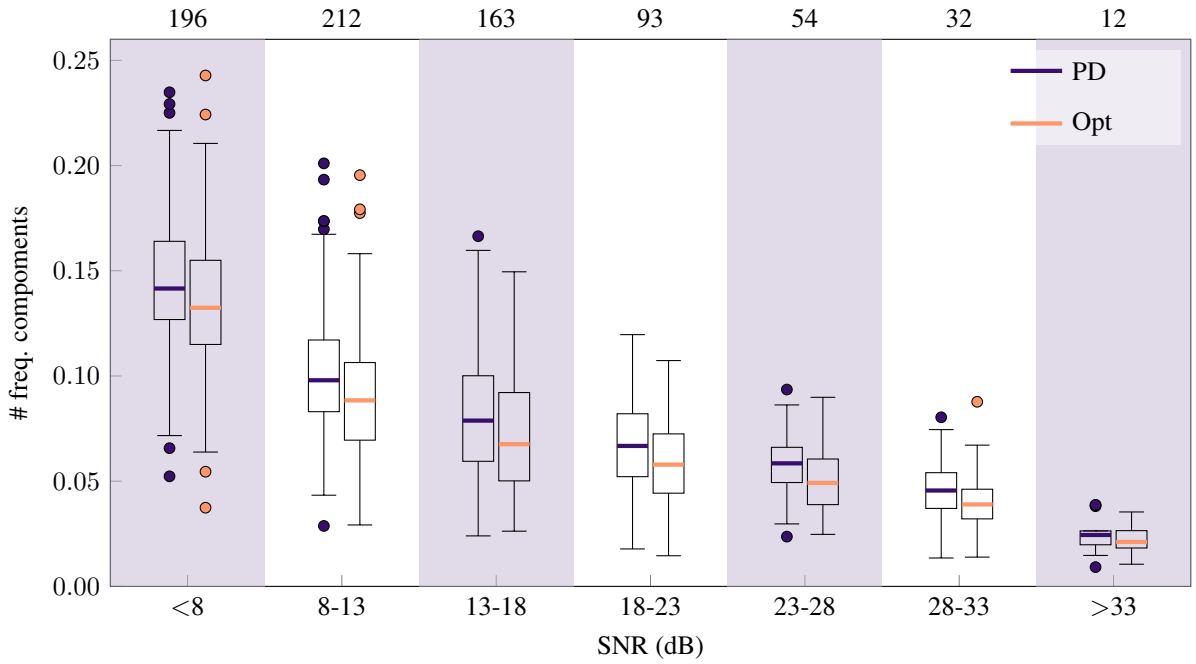


Figure 10.4: Box-Whiskers plot showing the NRMSD of the 2d system matrices recovered from 6-fold undersampled data. The boxes accumulate the 25% to 75% quantile. The numbers on top of the SNR bins denote the number of frequency components contained in them.

relationship is illustrated in the bottom panel of Figure 10.5 where we plot the $\overline{\text{NRMSD}}$ in dependence of the undersampling factor.

10.5.2 3d System Matrices

For the 3d experiment, the non-uniform distribution of samples becomes even more apparent. As can be seen in Figure 10.6, the sampling density in the center of the FOV is increased whereas there is a reduced number of samples outside the DF-FOV. At the large undersampling factor considered in this example, this leads to a clear improvement of the recovery within the DF-FOV. This can be seen in the second and third row of Figure 10.7, The residual structures in the error maps for the random sampling indicate a bias which is greatly reduced in the recoveries using the optimized sampling patterns. On other hand, the first row shows a case where the overall recovery error is slightly increased compared to the PD pattern. This is most probably caused by the simple spatial structure, which extends further beyond the DF-FOV than many other frequency components.

As can be seen in Figure 10.8, optimizing sampling patterns involves a trade-off caused by the different structures of the frequency components at hand. More precisely, it can be observed that the optimized sampling patterns lead to an increased NRMSD compared to the random sampling patterns for frequency components with high SNR. In contrast, the NRMSD is reduced for patterns with SNR $< 16\,\text{dB}$. Taking

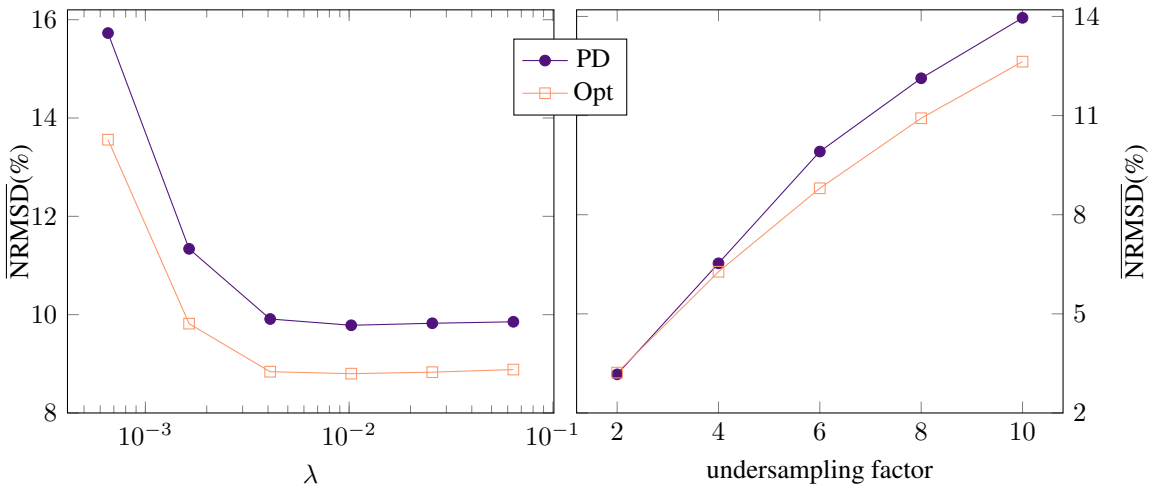


Figure 10.5: $\overline{\text{NRMSD}}$ of the 2d system matrices in dependence of the regularization parameter λ for 6-fold undersampling (top) and $\overline{\text{NRMSD}}$ in dependence of the undersampling factor (bottom). For each undersampling factor, λ was chosen such that the $\overline{\text{NRMSD}}$ was minimized for the respective method.

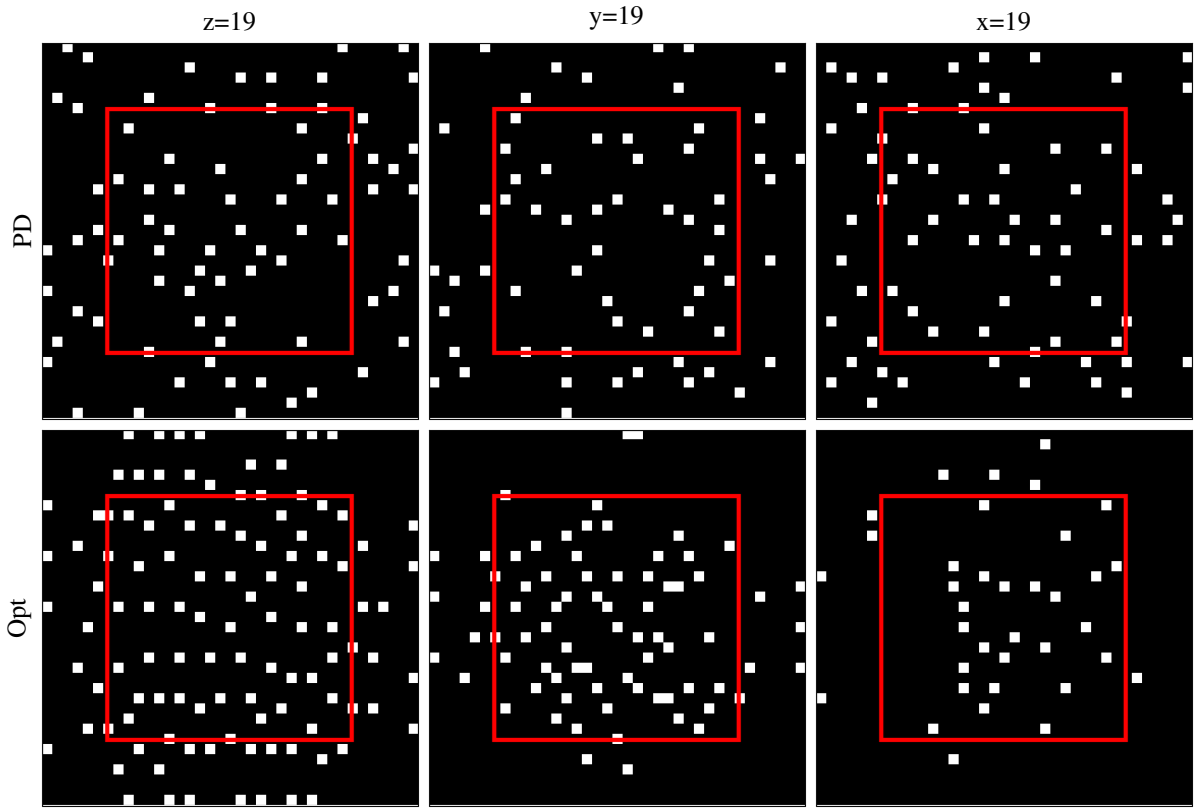


Figure 10.6: Sampling patterns generated using Poisson disk sampling (top row) and the proposed method (bottom row) for an undersampling factor of 20. The columns show cuts in the xy-, xz- and yz-plane respectively. For reference, the borders of the DF-FOV are indicated by red lines.

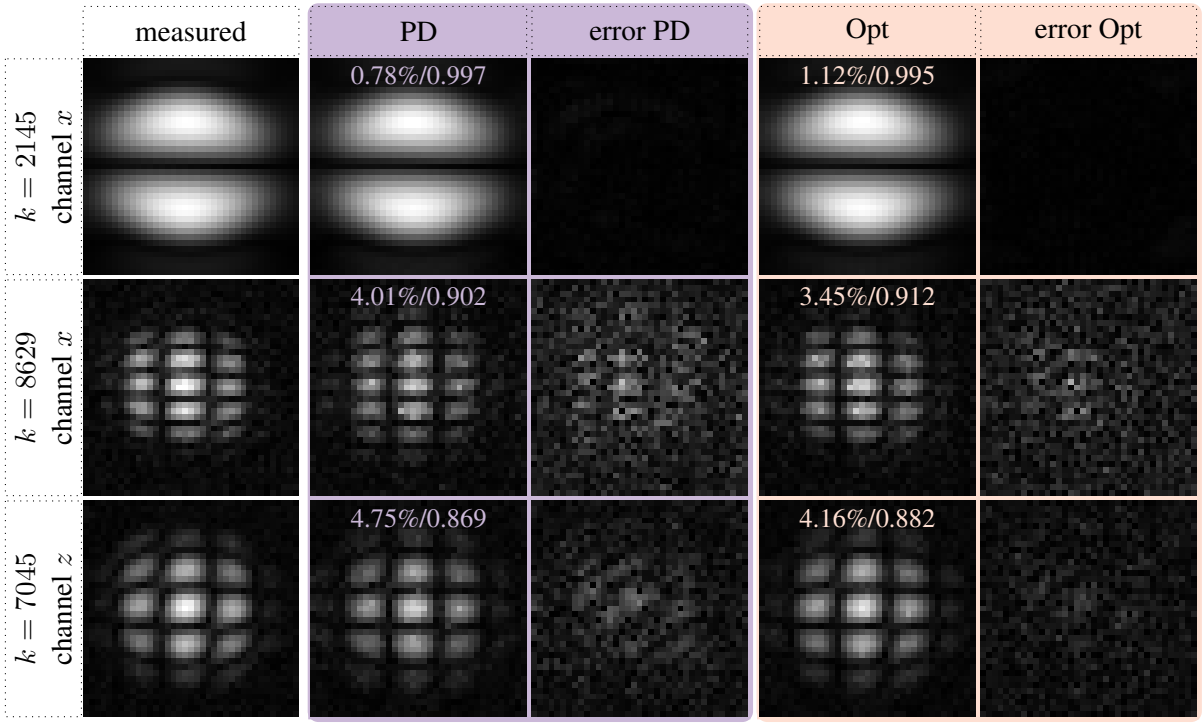


Figure 10.7: Recovered frequency components from 20-fold undersampled data. The second and third column show the patterns obtained from Poisson disk sampling and the respective error plots. The fourth and fifth column show the corresponding quantities for patterns obtained using Algorithm 5. The NRMSD (in %) and the SSIM with respect o the fully sampled reference are superimposed onto the plots. All plots in a given row are scaled to the same colorbar.

into account all frequency components, the $\overline{\text{NRMSD}}$ is 5.37 % for the random sampling pattern and 5.00 % for the optimized sampling pattern. The corresponding values for the $\overline{\text{SSIM}}$ are 0.838 for the random sampling pattern and 0.849 for the optimized sampling pattern.

The importance of accurately recovering the frequency components with low SNR becomes evident, when using the recovered system matrices for image reconstruction. The images obtained from the randomly sampled system matrix contain an increased amount of noise, as can be seen in Figure 10.9. For the optimized patterns, the noise is clearly reduced. This improvement is also reflected in the NRMSD and SSIM values, which are stated below the respective reconstructions.

10.6 Discussion

Our results show that the CS based recovery of MPI system matrices can indeed be enhanced by using sampling patterns optimized based on the Bayesian FIM. In contrast to pseudo-random sampling patterns, the optimized ones display a non-uniform sampling density, which is caused by the increased signal intensity typically observed in the FOV center as opposed to the borders of the FOV. In the context

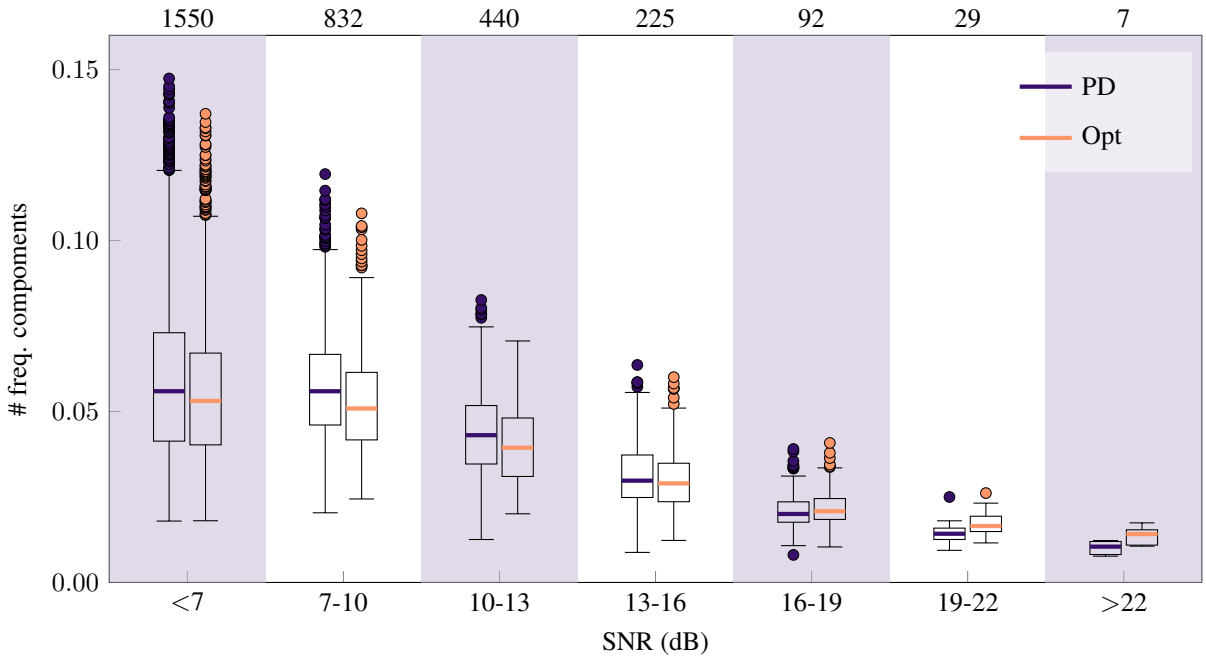


Figure 10.8: Box-Whiskers plot showing the NRMSD of the 3d system matrices recovered from 6-fold undersampled data. The boxes accumulate the 25% to 75% quantile. The numbers on top of the SNR bins denote the number of frequency components contained in them.

of MPI system matrix recovery, this constitutes the first work investigating variable density sampling schemes.

In all of our tests, the optimized sampling patterns resulted in an improved recovery of the system matrices used for testing in terms of the $\overline{\text{NRMSD}}$. A more detailed look reveals that the optimized patterns can lead to a slightly worse recovery of frequency components with high SNR, whereas the recovery is clearly improved for patterns with low SNR. An explanation is that frequency components with high SNR are often associated with low mixing factors and a slow signal decay outside the DF-FOV. On the other hand, the frequency components with low SNR often have a higher mixing order and a stronger signal decay outside the DF-FOV. The latter frequency components profit more from the denser sampling of the DF-FOV, whereas the former are recovered more efficiently using more uniform sampling patterns. Overall it should be noted that the bins with small SNR contain a lot more frequency components than the ones with high SNR, thus explaining the overall improvement in $\overline{\text{NRMSD}}$.

When using the recovered frequency components for image reconstruction, one should keep in mind that the frequency components with high mixing order play an important role whenever the image to be reconstructed contains a significant amount of high-resolution features. In view of our previous discussion, this explains the improvement in image quality to be observed in our image reconstruction experiments. Moreover, it indicates that the proposed method can prove particularly useful when applied in a setting that targets high image resolution.

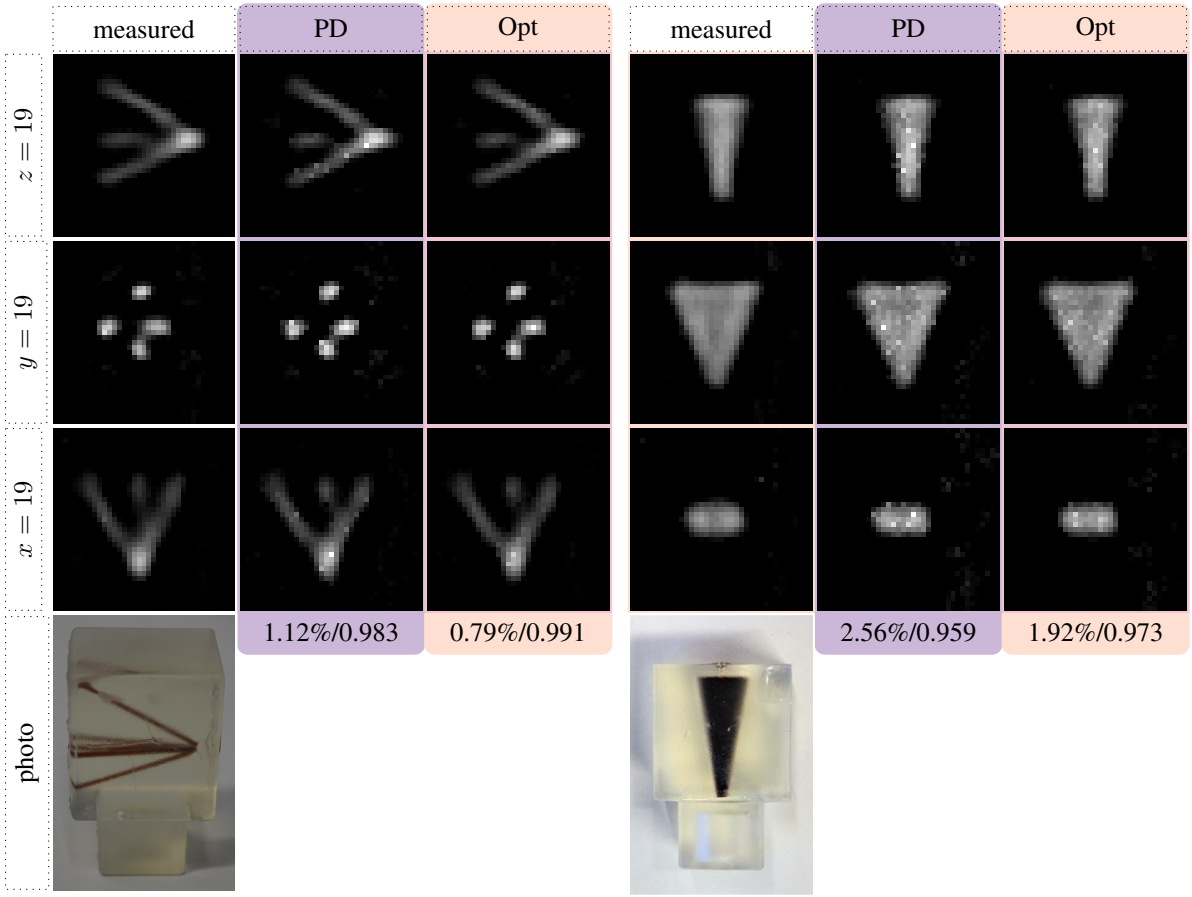


Figure 10.9: Reconstruction of the Open MPI Data resolution phantom (left panel) and the shape phantom (right panel). The columns show cuts in the xy-, xz-, and yz-plane respectively. The numbers below the reconstructions denote their NRMSD (in %) and their SSIM with respect to the reference reconstruction. All plots in a given row are scaled to the same colorbar.

In the proposed method, the parameter μ plays an important role as it controls the degree of non-uniformity of the obtained sampling patterns. A smaller value of μ typically leads to a higher degree of non-uniformity. In our experiments, a value of 3 led to good results. For smaller values, the decreased number of sampling points in the periphery of the FOV may lead to a reduced accuracy in the recovered system matrices, especially for frequency components with low mixing order.

An interesting aspect of the obtained sampling patterns is that overscanning the DF-FOV becomes inexpensive in terms of measurement time due to the reduced number of measurements outside the DF-FOV. Such an overscan of the DF-FOV is attractive because it reduces signal leakage from particles outside the DF-FOV [166]. When performing imaging of multiple patches, the overscan can be used to generate overlap between these patches. In turn, this will lead to an improved image reconstruction when using a suitable multi-patch reconstruction scheme.

The sampling patterns obtained with the proposed method depend on the sparsity structure of the system matrix rows used to setup problem (9.6). One can expect that the generated patterns will generalize well to other system matrices with a similar structure. Robustness can be increased by using a larger number of frequency components, which might even come from multiple system matrices. Concerning this robustness it is important to note that a significant part of the structure of MPI system matrices is determined by the field configuration and the imaging sequence used. Thus, it can be expected that system matrices measured with different tracers have similar sparsity structures. A further point to watch when using the proposed method is that the spatial dependence of the signal intensities, which is used in (10.1) should show a similar behavior for both the system matrix used for training and the one to be recovered. This spatial dependence is strongly depended on the magnetization curve of the particles.

Concerning aforementioned generalizability, our results illustrate that the adapted sampling patterns work quite well for the recovery of system matrices measured with other types of tracers. Importantly, our experiments include the case where training was performed using liquid particles and a system matrix measured with immobilized particles was recovered. It is well known that the transition from liquid to immobilized particles leads to a significant change of the spatial structure of system matrix patterns [126]. Nevertheless, the optimized sampling patterns resulted in an improved accuracy of the recovered system matrix in this challenging test case. For use in practical applications, we believe that a prior characterization of the particles to be used can help to make sure that a trained pattern is only applied to system matrices, whose structure is sufficiently similar to the training data. Setting up such a system will be the subject of a future study.

A potential alternative to the approach followed in this work, is to directly generate sampling patterns with a variable sampling density. Efficient algorithms for this task are known and routinely used in MRI and computer graphics [86, 87]. However, we note that such an approach would require extensive parameter tuning for choosing the sampling density. Moreover, a new pattern needs to be generated for every undersampling factor. In contrast, the proposed method determines the sampling density directly from the training data. Additionally, it yields an ordered list of samples to be measured. Thus, sampling patterns with varying undersampling factors can be generated simply by using the corresponding number of measurements from that list.

In conclusion, the described method can contribute to the acceleration of the calibration process in MPI and thus save valuable time. Due to the generic optimization based approach, we expect it to prove beneficial for other scanner architectures, such as single-sided MPI scanners, which features a fairly inhomogeneous coil sensitivity profile. Another interesting application is the measurement of system matrices using non-delta samples as proposed in [119]. Such approaches could probably be accelerated even more using the described experiment design framework. Finally, it would be interesting to extend the framework to the joint recovery of multiple-system matrices in a multi-patch setting. In such a setting, even larger undersampling factors could be achieved by exploiting the similarity between patches [177]. An extension of our method could help answer the question how to optimally sample the system matrices in such a setting.

11 Conclusion and Outlook

11.1 Summary

This work focuses on the magnetic field based imaging modalities MRI and MPI. These modalities are complementary to each other in the sense that the anatomical information provided by MRI is needed to properly interpret the functional information provided by MPI. Moreover both do not rely on ionizing radiation, which makes their combination particularly attractive.

The main objective of this work was the development of algorithmic solutions to the problem of long measurement times, which commonly arises during the acquisition of MRI imaging data and during the measurement of MPI system matrices. Interestingly, both of these problems have a very similar mathematical structure and can thus be tackled using similar techniques. In particular, recent works have demonstrated that prior knowledge, such as the transform-domain sparsity of images, can be used to reduce the number of measurements required for successful image reconstruction and thus to minimize measurement time. Despite the amount of research already performed, the application of CS in MRI and MPI still faces different algorithmic challenges. These include finding useful forms of prior knowledge to improve image quality and the development of efficient optimization methods. Finally, it is an open question how to best chose the measurements performed in an imaging experiment.

Principles of MRI and MPI

To tackle aforementioned challenges, Part I of this work introduced the basic mathematical and physical principles which underlie the imaging modalities MPI and MRI and their corresponding image reconstruction methods. In particular, chapter 2 described a generic mathematical framework that can be used to describe general imaging experiments and the associated image reconstruction methods. These mathematical basics are complemented by the physical concepts underlying MPI and MRI, which were discussed in chapter 3.

Making use of aforementioned principles, chapter 4 introduced a mathematical description of MRI imaging experiments based on the phenomenological Bloch equations. An important aspect of this chapter was to show how the mathematical framework from chapter 2 can be used to describe MRI experiments and how the most commonly used image reconstruction methods can be derived from it. In a

similar way, chapter 5 introduced the basic principles of MPI along with a mathematical description of the corresponding imaging experiments and the commonly used reconstruction methods. A further focus of this chapter was the structure and the measurement of the MPI system matrix as its recovery is one focus of this work.

Compressed Sensing

Part II of this work focused on the challenges arising when applying CS to the modalities MRI and MPI. A basic requirement for the development of image reconstruction algorithms is a software framework providing the required functionality for the loading and processing of measurement data as well as for the formulation and solution of the resulting image reconstruction problems. To address this need, chapter 6 described the MRI image reconstruction framework *MRIReco.jl* and its MPI counterpart *MPIReco.jl*. Both frameworks combine high performance and high ease-of-use. In contrast to other state-of-the-art MRI reconstruction frameworks, they are written in a single programming language, which makes them easy to use and greatly simplifies the process of extending and customizing them, thus facilitating algorithmic research. The numerical experiments in this chapter focused on *MRIReco.jl* and demonstrated that its performance is indeed on par with other state-of-the-art reconstruction frameworks. For *MPIReco.jl* no comparable benchmarks were performed because *MPIReco.jl* was the first open source MPI reconstruction frame work to be published. However, examples for its application are contained in the subsequent chapters on MPI system matrix recovery.

In MRI, the reconstruction of undersampled, non-Cartesian data poses a particular challenge. On the one hand, this is caused by the non-Cartesian sampling which often slows down the convergence of commonly used reconstruction methods. On the other hand, non-Cartesian sampling schemes often require a computationally expensive B_0-aware signal model in order to mitigate artifacts. This leads to a further significant increase in reconstruction time. To address these issues, chapter 7 proposed an SVD-based scheme, which allows to efficiently evaluate the B_0-aware signal model. In contrast to other methods, this scheme allows to tailor the numerical complexity of the used approximation to the structure of the given field map, which is important to avoid performing unnecessary computations. To combat aforementioned convergence issues, the proposed signal model was combined with a diagonal k-space preconditioner, which significantly reduced the number of iterations required for successful image reconstruction. While this preconditioning scheme was originally proposed in Ref. [43], one important aspect of this work was its extension and validation for the case where B_0 inhomogeneities can no longer be neglected.

When considering the recovery of MPI system matrices, computation time is less crucial. However, previous approaches have all relied on a simple sparsity model in the Fourier domain. To improve upon this, chapter 8 introduced a representation of the MPI system matrix rows as low-rank tensors. This low-rank representation is a powerful prior which complements the existing sparsity-based ones. To exploit it, the system matrix recovery problem was extended and it was shown how the resulting inverse problem can be solved using the Split Bregman method. Numerical experiments showed that the proposed method

yields system matrices with increased quality and significantly reduced noise compared to the standard CS method.

Optimized Experiment Design

Finally, Part III of this work investigated the question how the measurements performed in a CS-based imaging scheme can be optimized for a given class of images. In the past, the measurements to be performed were often chosen according to heuristics motivated by the theory of compressed sensing and traditional sampling theory. While being simple to implement, these sampling schemes often do not lead to optimal reconstruction results. A more promising approach is to tailor sampling patterns to the structure of the known image class. To generate such optimized sampling patterns, chapter 9 proposed a framework which aims to minimize the expected reconstruction error as quantified by the Bayesian Fisher information matrix. Moreover, an algorithm based on the sequential forward selection method was proposed to efficiently solve the corresponding optimization problem. To validate it, the method was applied for the generation of MRI sampling patterns. The resulting patterns yielded improved image reconstruction results when compared to popularly used pseudo-random sampling schemes. Additionally, the computation time required for generation of the sampling patterns was significantly reduced in comparison to previously proposed methods.

Chapter 10 demonstrated how the previously described framework can be applied to optimize sampling patterns for the measurement of MPI system matrices. The Bayesian nature of the framework allowed us to encode prior information both about the sparsity structure of the system matrix rows and about the spatially dependent distribution of signal energy. In contrast to previously used sampling patterns, the optimized sampling patterns showed a variable density with a higher number of samples in the center of the FOV and a correspondingly reduced sampling density near the FOV boundaries. When used for system matrix recovery, the optimized schemes led to a visible improvement of the recovered system matrices, which was also reflected in the quality of subsequently reconstructed imaging data.

11.2 Outlook

This work addressed a fairly broad range of challenges, which arise when using CS to accelerate the acquisition of MRI data and MPI system matrices. Nevertheless, there remain more challenges to be addressed in order to further improve the application of CS-acquisition schemes in MRI and MPI.

While *MRIReco.jl* and *MPIReco.jl* proved highly useful for the development of the reconstruction algorithms presented in this work, we note that they are not finished products and that further development is to be expected. Especially for *MRIReco.jl*, it would be beneficial to provide an even larger range of pre- and post-processing techniques that are often used in MRI in order to facilitate a wide adoption of the framework. Moreover, some common acquisition schemes such as partial Fourier acquisitions are not fully supported yet.

Turning to the B_0-aware reconstruction of MRI data, we note that the determination of the field map itself is a challenge, which was not considered in this work. Commonly, the field map is determined using a separated calibration scan. However, this can lead to problems as it does not capture dynamic B_0 variations, which are induced e.g. by patient movements. An interesting approach to circumvent this issue is to jointly estimate the field map along with the imaging data using approaches similar to the ones proposed in [79, 81, 178].

The proposed low-rank representation of the MPI system matrix greatly shows how prior knowledge can enhance system matrix recovery. Further improvements can be expected when more of the inherent structures of the MPI system matrix are exploited. Up to this point, system matrix rows are recovered independently from each other. Thus, any correlations between different system matrix rows are neglected. Taking into account such inter-row-correlations is thus expected to yield further improvements. In a similar way, it would be interesting to exploit the similarity between the multiple system matrices required to perform multi-patch imaging. Such inter-patch similarities can be exploited using low-rank regularization as was explored in the work presented in [C5].

The proposed method for designing sampling patterns has shown promising results both for MRI and MPI. Nevertheless, we expect further improvements to be possible especially for the case of multi-contrast MRI and multi-patch MPI. For both cases, it would be highly desirable to exploit inter-contrast and inter-patch correlations in the design of the sampling patterns. This could yield further improvements especially when paired with suitable reconstruction schemes. Another interesting avenue for future research is to consider more general sampling schemes. For instance, MRI measurements could be significantly accelerated by optimizing the shape of the used sampling trajectories for a given signal class as was recently proposed in [179]. For the MPI system matrix, it would be interesting to perform system matrix measurements using extended samples, such as the coded calibration scenes proposed in [119], as this could lead to a further boost in signal-to-noise ratio.

Finally, we note that the present work mainly focused on classical optimization-based methods which require no or only very small sets of training data. In recent years, machine learning models trained on large datasets have shown a lot of potential to further improve CS-based image reconstruction [C9, 89–91]. A drawback of these methods is that most machine-learning models rely on large training datasets and that their robustness to unseen data requires careful investigation before such models can be deployed in practice. With this in mind, especially the combination of classical optimization methods with learning-based models are great candidates to further improve upon the methods proposed in this work.

Bibliography

Own Publications

Journal Publications

[J1] M. Grosser, M. Möddel, and T. Knopp, "Using Low-Rank Tensors for the Recovery of MPI System Matrices," *IEEE Trans. Comput. Imaging*, vol. 6, no. 1, pp. 1389–1402, 2020. DOI: 10.1109/TCI.2020.3024078.

[J2] M. Grosser and T. Knopp, "Optimized sampling patterns for the sparse recovery of system matrices in Magnetic Particle Imaging," *Int. J. Magn. Part. Imaging*, vol. 7, no. 2, 2112001, 2021. DOI: 10.18416/IJMPI.2021.2112001.

[J3] T. Knopp and M. Grosser, "MRIReco.jl: An MRI reconstruction framework written in Julia," *Magn. Reson. Med.*, vol. 86, no. 3, pp. 1633–1646, 2021. DOI: 10.1002/mrm.28792.

[J4] T. Knopp, P. Szwargulski, F. Griese, M. Grosser, M. Boberg, and M. Möddel, "MPIReco.jl: Julia package for image reconstruction in MPI," *Int. J. Magn. Part. Imaging*, vol. 5, no. 1-2, 1907001, 2019. DOI: 10.18416/IJMPI.2019.1907001.

[J5] T. Knopp, M. Grosser, M. Graeser, T. Gerkmann, and M. Möddel, "Efficient Joint Estimation of Tracer Distribution and Background Signals in Magnetic Particle Imaging using a Dictionary Approach," *IEEE Trans. Med. Imaging*, vol. 40, no. 12, pp. 3568–3579, 2021. DOI: 10.1109/TMI.2021.3090928.

[J6] I. Molwitz, G. M. Campbell, J. Yamamura, T. Knopp, K. Toedter, R. Fischer, Z. J. Wang, A. Busch, A.-K. Ozga, S. Zhang, T. Lindner, S. F., M. Grosser, G. Adam, and P. Szwargulski, "Fat Quantification in Dual-Layer Detector Spectral Computed Tomography: Experimental Development and First In-Patient Validation," *Investig. Radiol.*, vol. 57, no. 7, pp. 463–469, 2022. DOI: 10.1097/RLI.0000000000000858.

[J7] T. Knopp, M. Boberg, and M. Grosser, "NFFT. jl: Generic and Fast Julia Implementation of the Nonequidistant Fast Fourier Transform," *SIAM J. Sci. Comput. (accepted)*, 2023, preprint available at: https://arxiv.org/abs/2208.00049.

Conference Proceedings

[C1] M. Grosser and T. Knopp, "Efficient Optimization Of MRI Sampling Patterns Using The Bayesian Fisher Information Matrix," in *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, 2021, pp. 234–237. DOI: 10.1109/ISBI48211.2021.9434109.

[C2] M. Grosser and T. Knopp, "Low rank approach to sparse system matrix recovery for MPI," in *Int. Workshop Magn. Part. Imaging Book Abstr.*, vol. 9, 2019.

[C3] M. Grosser and T. Knopp, "Fast Image Reconstruction for Non-Cartesian Acquisitions in the Presence of B0-inhomogeneities," in *Proc. Int. Soc. Magn. Reson. Med.*, vol. 29, 2021, 3525.

[C4] M. Grosser and T. Knopp, "Accelerated Multi-Echo Gradient Echo Imaging using Locally-Low Rank Regularization," in *Proc. Int. Soc. Magn. Reson. Med.*, vol. 28, 2020, 3732.

[C5] M. Grosser, M. Boberg, M. Bahe, and T. Knopp, "Enhanced compressed sensing recovery of multi-patch system matrices in MPI," in *Int. J. Magn. Part. Imaging*, vol. 6, 2020, 2009035. DOI: 10.18416/IJMPI.2020.2009035.

[C6] T. Knopp and M. Grosser, "MRIReco.jl: An extensible Open-Source Image Reconstruction Framework written in Julia," in *Proc. Int. Soc. Magn. Reson. Med.*, vol. 27, 2019, 4835.

[C7] T. Knopp, M. Grosser, M. Graeser, T. Gerkmann, and M. Möddel, "Dictionary-Based Background Signal Estimation For Magnetic Particle Imaging," in *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, 2021, pp. 1540–1543. DOI: 10.1109/ISBI48211.2021.9434048.

[C8] I. M. Baltruschat, P. Szwargulski, F. Griese, M. Grosser, R. Werner, and T. Knopp, "Reduktion der Kalibrierungszeit für die Magnetpartikelbildgebung mittels Deep Learning," in *Bildverarbeitung für die Medizin 2021*, 2021, pp. 337–337. DOI: 10.1007/978-3-658-33198-6_82.

[C9] I. M. Baltruschat, P. Szwargulski, F. Griese, M. Grosser, R. Werner, and T. Knopp, "3d-SMRnet: Achieving a new quality of MPI system matrix recovery by deep learning," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, 2020, pp. 74–82. DOI: 10.1007/978-3-030-59713-9_8.

[C10] L. Nawwas, M. Grosser, M. Möddel, and T. Knopp, "Accelerated Kaczmarz for Convergence Speed-up in Multi-Contrast Magnetic Particle Imaging," in *Int. J. Magn. Part. Imaging*, vol. 8, 2022, 203022. DOI: 10.18416/ijmpi.2022.2203022.

[C11] T. Knopp and M. Grosser, "Warmstart Approach for Accelerating Deep Image Prior Reconstruction in Dynamic Tomography," in *Proceedings of The 5th International Conference on Medical Imaging with Deep Learning*, 2022, pp. 713–725.

General Publications

[1] T. M. Buzug, *Computed Tomography: From Photon Statistics to Modern Cone-Beam CT*. Berlin / Heidelberg: Springer, 2008.

[2] O. Dössel, *Bildgebende Verfahren in der Medizin*. Heidelberg: Springer, 2000.

[3] Z.-P. Liang and P. Lauterbur, *Principles of magnetic resonance imaging: a signal processing perspective*. New York: Wiley-IEEE Press, 2000.

[4] D. Bryant, J. Payne, D. Firmin, and D. B. Longmore, "Measurement of flow with NMR imaging using a gradient pulse and phase difference technique," *J. Comput. Assist. Tomogr.*, vol. 8, no. 4, pp. 588–593, 1984. DOI: 10.1097/00004728-198408000-00002.

[5] D. Le Bihan, E. Breton, D. Lallemand, P. Grenier, E. Cabanis, and M. Laval-Jeantet, "MR imaging of intravoxel incoherent motions: application to diffusion and perfusion in neurologic disorders," *Radiology*, vol. 161, no. 2, pp. 401–407, 1986. DOI: 10.1148/radiology.161.2.3763909.

[6] S. Ogawa, T.-M. Lee, A. R. Kay, and D. W. Tank, "Brain magnetic resonance imaging with contrast dependent on blood oxygenation," *Proc. Natl. Acad. Sci.*, vol. 87, no. 24, pp. 9868–9872, 1990. DOI: 10.1073/pnas.87.24.9868.

[7] S. Ogawa, D. W. Tank, R. Menon, J. M. Ellermann, S. G. Kim, H. Merkle, and K. Ugurbil, "Intrinsic signal changes accompanying sensory stimulation: Functional brain mapping with magnetic resonance imaging," *Proc. Natl. Acad. Sci.*, vol. 89, no. 13, pp. 5951–5955, 1992. DOI: 10.1073/pnas.89.13.5951.

[8] F. Schweser, K. Sommer, A. Deistung, and J. R. Reichenbach, "Quantitative susceptibility mapping for investigating subtle susceptibility variations in the human brain," *Neuroimage*, vol. 62, no. 3, pp. 2083–2100, 2012. DOI: 10.1016/j.neuroimage.2012.05.067.

[9] B. Gleich and J. Weizenecker, "Tomographic imaging using the nonlinear response of magnetic particles," *Nature*, vol. 435, no. 7046, pp. 1214–1217, 2005. DOI: 10.1038/nature03808.

[10] J. Weizenecker, B. Gleich, J. Rahmer, H. Dahnke, and J. B.gert, "Three-dimensional real-time in vivo magnetic particle imaging," *Phys. Med. Biol.*, vol. 54, no. 5, pp. L1–L10, 2009. DOI: 10.1088/0031-9155/54/5/l01.

[11] M. Kaul, J. Salamon, T. Knopp, H. Ittrich, G. Adam, H. Weller, and C. Jung, "Magnetic particle imaging for in vivo blood flow velocity measurements in mice," *Phys. Med. Biol.*, vol. 63, no. 6, 064001, 2018. DOI: 10.1088/1361-6560/aab136.

[12] D. Pantke, F. Mueller, S. Reinartz, F. Kiessling, and V. Schulz, "Flow velocity quantification by exploiting the principles of the Doppler effect and magnetic particle imaging," *Sci. Rep.*, vol. 11, no. 1, pp. 1–15, 2021. DOI: 10.1038/s41598-021-83821-w.

[13] P. Szwargulski, M. Wilmes, E. Javidi, F. Thieben, M. Graeser, M. Koch, C. Gruettner, G. Adam, C. Gerloff, T. Magnus, *et al.*, "Monitoring Intracranial Cerebral Hemorrhage Using Multicontrast Real-Time Magnetic Particle Imaging," *ACS Nano*, vol. 14, no. 10, pp. 13 913–13 923, 2020. DOI: 10.1021/acsnano.0c06326.

[14] J. Haegele, N. Panagiotopoulos, S. Cremers, J. Rahmer, J. Franke, R. Duschka, A. Vaalma, M. Heidenreich, J. Borgert, P. Borm, *et al.*, "Magnetic particle imaging: A resovist based marking technology for guide wires and catheters for vascular interventions," *IEEE Trans. Med. Imaging*, vol. 35, no. 10, pp. 2312–2318, 2016. DOI: 10.1109/TMI.2016.2559538.

[15] J. Haegele, S. Biederer, H. Wojczyk, M. Gräser, T. Knopp, T. Buzug, J. Barkhausen, and F. Vogt, "Towards cadiovascular interventions guided by magnetic particle imaging: First instrument characterization," *Magn. Reson. Med.*, vol. 69, no. 6, pp. 1761–1767, 2013. DOI: 10.1002/mrm.24421.

[16] P. Ludewig, N. Gdaniec, J. Sedlacik, N. D. Forkert, P. Szwargulski, M. Graeser, G. Adam, M. Kaul, K. Krishnan, R. Ferguson, *et al.*, "Magnetic Particle Imaging for Real-Time Perfusion Imaging in Acute Stroke," *ACS Nano*, vol. 11, no. 10, pp. 10 480–10 488, 2017. DOI: 10.1021/acsnano.7b05784.

[17] J. Franke, U. Heinen, H. Lehr, A. Weber, F. Jaspard, W. Ruhm, M. Heidenreich, and V. Schulz, "System Characterization of a Highly Integrated Preclinical Hybrid MPI-MRI Scanner," *IEEE Trans. Med. Imaging*, vol. 35, no. 9, pp. 1993–2004, 2016. DOI: 10.1109/TMI.2016.2542041.

[18] C. E. Shannon, "Communication in the presence of noise," *Proc. IRE*, vol. 37, no. 1, pp. 10–21, 1949. DOI: 10.1109/JRPROC.1949.232969.

[19] E. J. Candes, J. K. Romberg, and T. Tao, "Stable signal recovery from incomplete and inaccurate measurements," *Commun. Pure Appl. Math.*, vol. 59, no. 8, pp. 1207–1223, 2006. DOI: 10.1002/cpa.20124.

[20] D. L. Donoho, "Compressed sensing," *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289–1306, 2006. DOI: 10.1109/TIT.2006.871582.

[21] E. J. Candes and J. K. Romberg, "Sparsity and incoherence in compressive sampling," *Inverse Probl.*, vol. 23, no. 3, 969, 2007. DOI: 10.1088/0266-5611/23/3/008.

[22] Y. Oike and A. El Gamal, "CMOS image sensor with per-column $\Sigma\Delta$ ADC and programmable compressed sensing," *IEEE J. Solid-State Circuits*, vol. 48, no. 1, pp. 318–328, 2012. DOI: 10.1109/JSSC.2012.2214851.

[23] Y. Wiaux, L. Jacques, G. Puy, A. M. M. Scaife, and P. Vandergheynst, "Compressed sensing imaging techniques for radio interferometry," *Mon. Not. R. Astron. Soc.*, vol. 395, no. 3, pp. 1733–1742, 2009. DOI: 10.1111/j.1365-2966.2009.14665.x.

[24] M. H. Firooz and S. Roy, "Network tomography via compressed sensing," in *IEEE Global Telecommunications Conference GLOBECOM 2010*, 2010, pp. 1–5. DOI: 10.1109/GLOCOM.2010.5684036.

[25] S. Krönke, J. Knörzer, and P. Schmelcher, "Correlated quantum dynamics of a single atom collisionally coupled to an ultracold finite bosonic ensemble," *New J. Phys.*, vol. 17, no. 5, 053001, 2015. DOI: 10.1088/1367-2630/17/5/053001.

[26] M. Lustig, D. Donoho, and J. M. Pauly, "Sparse MRI: The application of compressed sensing for rapid MR imaging," *Magn. Reson. Med.*, vol. 58, no. 6, pp. 1182–1195, 2007. DOI: 10.1002/mrm.21391.

[27] M. Lustig and J. M. Pauly, "SPIRiT: iterative self-consistent parallel imaging reconstruction from arbitrary k-space," *Magn. Reson. Med.*, vol. 64, no. 2, pp. 457–471, 2010. DOI: 10.1002/mrm.22428.

[28] T. Knopp and A. Weber, "Sparse reconstruction of the magnetic particle imaging system matrix," *IEEE Trans. Med. Imaging*, vol. 32, no. 8, pp. 1473–1480, 2013. DOI: 10.1109/TMI.2013.2258029.

[29] A. Weber and T. Knopp, "Reconstruction of the magnetic particle imaging system matrix using symmetries and compressed sensing," *Adv. Math. Phys.*, vol. 2015, 460496, 2015. DOI: 10.1155/2015/460496.

[30] S. Pejoski, V. Kafedziski, and D. Gleich, "Compressed sensing MRI using discrete nonseparable shearlet transform and FISTA," *IEEE Signal Process. Lett.*, vol. 22, no. 10, pp. 1566–1570, 2015. DOI: 10.1109/LSP.2015.2414443.

[31] S. Ravishankar and Y. Bresler, "MR image reconstruction from highly undersampled k-space data by dictionary learning," *IEEE Trans. Med. Imaging*, vol. 30, no. 5, pp. 1028–1041, 2010. DOI: 10.1109/TMI.2010.2090538.

[32] S. Ravishankar and Y. Bresler, "Data-driven learning of a union of sparsifying transforms model for blind compressed sensing," *IEEE Trans. Comput. Imaging*, vol. 2, no. 3, pp. 294–309, 2016. DOI: 10.1109/TCI.2016.2567299.

[33] J. P. Haldar and D. Kim, "OEDIPUS: An experiment design framework for sparsity-constrained MRI," *IEEE Trans. Med. Imaging*, vol. 38, no. 7, pp. 1545–1558, 2019. DOI: 10.1109/TMI.2019.2896180.

[34] M. Seeger, H. Nickisch, R. Pohmann, and B. Schölkopf, "Optimization of k-space trajectories for compressed sensing by Bayesian experimental design," *Magn. Reson. Med.*, vol. 63, no. 1, pp. 116–126, 2010. DOI: 10.1002/mrm.22180.

[35] E. Levine and B. Hargreaves, "On-the-Fly Adaptive k-Space Sampling for Linear MRI Reconstruction Using Moment-Based Spectral Analysis," *IEEE Trans. Med. Imaging*, vol. 37, no. 2, pp. 557–567, 2018. DOI: 10.1109/TMI.2017.2766131.

[36] F. Sherry, M. Benning, J. C. De Los Reyes, M. J. Graves, G. Maierhofer, G. Williams, C. B. Schönlieb, and M. J. Ehrhardt, "Learning the Sampling Pattern for MRI," *IEEE Trans. Med. Imaging*, vol. 39, no. 12, pp. 4310–4321, 2020. DOI: 10.1109/TMI.2020.3017353.

[37] S. Van Der Walt, S. C. Colbert, and G. Varoquaux, "The NumPy array: A structure for efficient numerical computation," *Comput. Sci. Eng.*, vol. 13, no. 2, pp. 22–30, 2011. DOI: 10.1109/MCSE.2011.37.

[38] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga, *et al.*, "PyTorch: An Imperative Style, High-Performance Deep Learning Library," in *Adv. Neural Inf. Process. Syst.*, vol. 32, 2019.

[39] M. S. Hansen and T. S. Sørensen, "Gadgetron: An open source framework for medical image reconstruction," *Magn. Reson. Med.*, vol. 69, no. 6, pp. 1768–1776, 2013. DOI: 10.1002/MRM.24389.

[40] M. Uecker, F. Ong, J. I. Tamir, D. Bahri, P. Virtue, J. Y. Cheng, T. Thang, and M. Lustig, "Berkeley advanced reconstruction toolbox," in *Proc. Int. Soc. Magn. Reson. Med.*, vol. 23, 2015, 2486.

[41] J. Bezanson, A. Edelman, S. Karpinski, and V. B. Shah, "Julia: A Fresh Approach to Numerical Computing," *SIAM Rev.*, vol. 59, no. 1, pp. 65–98, 2017. DOI: 10.1137/141000671.

[42] J. A. Fessler, S. Lee, V. T. Olafsson, H. R. Shi, and D. C. Noll, "Toeplitz-based iterative image reconstruction for MRI with correction for magnetic field inhomogeneity," *IEEE Trans. Signal Process.*, vol. 53, no. 9, pp. 3393–3402, 2005. DOI: 10.1109/TSP.2005.853152.

[43] F. Ong, M. Uecker, and M. Lustig, "Accelerating non-Cartesian MRI reconstruction convergence using k-space preconditioning," *IEEE Trans. Med. Imaging*, vol. 39, no. 5, pp. 1646–1654, 2019. DOI: 10.1109/TMI.2019.2954121.

[44] K. P. Pruessmann, M. Weiger, P. Börnert, and P. Boesiger, "Advances in sensitivity encoding with arbitrary k-space trajectories," *Magn. Reson. Med*, vol. 46, no. 4, pp. 638–651, 2001. DOI: https://doi.org/10.1002/mrm.1241.

[45] G. H. Golub and C. F. Van Loan, *Matrix computations, 4th*. Baltimore: Johns Hopkins University Press, 2013.

[46] D. C. Montgomery and G. C. Runger, "Applied statistics and probability for engineers," in Hoboken: John Wiley & Sons, 2014, ch. 7, pp. 241–247.

[47] S. M. Kay, "Fundamentals of Statistical Signal Processing: Estimation Theory," in New Jersey: Prentice Hall PTR, 1993, ch. 12, pp. 379–418.

[48] R. Penrose, "A generalized inverse for matrices," *Math. Proc. Camb. Philos. Soc.*, vol. 51, no. 3, pp. 406–413, 1955. DOI: 10.1017/S0305004100030401.

[49] D. Taubman and M. Marcellin, *JPEG2000 image compression fundamentals, standards and practice: image compression fundamentals, standards and practice* (The Springer International Series in Engineering and Computer Science). Heidelberg: Springer Science & Business Media, 2012, vol. 642.

[50] D. L. Donoho and X. Huo, "Uncertainty principles and ideal atomic decomposition," *IEEE Trans. Inf. Theory*, vol. 47, no. 7, pp. 2845–2862, 2001. DOI: 10.1109/18.959265.

[51] D. L. Donoho, "For most large underdetermined systems of linear equations the minimal ℓ_1-norm solution is also the sparsest solution," *Commun. Pure Appl. Math.*, vol. 59, no. 6, pp. 797–829, 2006. DOI: 10.1002/cpa.20132.

[52] T. Goldstein and S. Osher, "The split Bregman method for L1-regularized problems," *SIAM J. Imaging Sci.*, vol. 2, no. 2, pp. 323–343, 2009. DOI: 10.1137/080725891.

[53] A. Antoniou and W.-S. Lu, "Practical optimization: Algorithms and engineering applications," in Heidelberg: Springer, 2007, ch. 6, pp. 145–171.

[54] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM J. Imaging Sci.*, vol. 2, no. 1, pp. 183–202, 2009. DOI: 10.1137/080716542.

[55] A. Chambolle and T. Pock, "A first-order primal-dual algorithm for convex problems with applications to imaging," *J. Math. Imaging Vis.*, vol. 40, no. 1, pp. 120–145, 2011. DOI: 10.1007/s10851-010-0251-1.

[56] M. Doneva, P. Börnert, H. Eggers, C. Stehning, J. Sénégas, and A. Mertins, "Compressed sensing reconstruction for magnetic resonance parameter mapping," *Magn. Reson. Med.*, vol. 64, no. 4, pp. 1114–1120, 2010. DOI: 10.1002/mrm.22483.

[57] K. Koolstra, J. van Gemert, P. Börnert, A. Webb, and R. Remis, "Accelerating compressed sensing in parallel imaging reconstructions using an efficient circulant preconditioner for cartesian trajectories," *Magn. Reson. Med.*, vol. 81, no. 1, pp. 670–685, 2019. DOI: 10.1002/MRM.27371.

[58] J. Trzasko, A. Manduca, and E. Borisch, "Local versus global low-rank promotion in dynamic MRI series reconstruction," in *Proc. Int. Soc. Magn. Reson. Med*, vol. 19, 2011, p. 4371.

[59] R. Otazo, E. Candes, and D. K. Sodickson, "Low-rank plus sparse matrix decomposition for accelerated dynamic MRI with separation of background and dynamic components," *Magn. Reson. Med.*, vol. 73, no. 3, pp. 1125–1136, 2015. DOI: 10.1002/mrm.25240.

[60] J. I. Tamir, M. Uecker, W. Chen, P. Lai, M. T. Alley, S. S. Vasanawala, and M. Lustig, "T2 shuffling: Sharp, multicontrast, volumetric fast spin-echo imaging," *Magn. Reson. Med.*, vol. 77, no. 1, pp. 180–195, 2017. DOI: 10.1002/mrm.26102.

[61] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, 2004. DOI: 10.1109/TIP.2003.819861.

[62] D. Brunet, E. R. Vrscay, and Z. Wang, "On the mathematical properties of the structural similarity index," *IEEE Trans. Image Process.*, vol. 21, no. 4, pp. 1488–1499, 2011. DOI: 10.1109/TIP.2011.2173206.

[63] J. D. Jackson, "Classical Electrodynamics," in New York: Wiley, 1998, ch. 6, pp. 170–179.

[64] T. Fließbach, *Elektrodynamik*. Heidelberg: Springer Spektrum, 2012.

[65] C. Kittel, "Introduction to Solid State Physics," in New York: Wiley, 2005, ch. 12, pp. 321–356.

[66] J. Hennig, "Echoes—how to generate, recognize, use or avoid them in MR-imaging sequences. Part I: Fundamental and not so fundamental properties of spin echoes," *Concepts Magn. Reson*, vol. 3, no. 3, pp. 125–143, 1991. DOI: 10.1002/cmr.1820030302.

[67] M. F. Reiser, W. Semmler, and H. Hricak, *Magnetic resonance tomography*. Heidelberg: Springer Science & Business Media, 2007.

[68] D. Weishaupt, V. D. Köchli, B. Marincek, J. M. Froehlich, D. Nanz, and K. P. Pruessmann, *How does MRI work?: an introduction to the physics and function of magnetic resonance imaging*. Heidelberg: Springer, 2006, vol. 2.

[69] K. P. Pruessmann, M. Weiger, M. Scheidegger, and P. Boesiger, "SENSE: sensitivity encoding for fast MRI," *Magn. Reson. Med.*, vol. 42, no. 5, pp. 952–962, 1999. DOI: 10.1002/(SICI)1522-2594(199911)42:5<952::AID-MRM16>3.0.CO;2-S.

[70] M. A. Griswold, P. M. Jakob, R. M. Heidemann, M. Nittka, V. Jellus, J. Wang, B. Kiefer, and A. Haase, "Generalized autocalibrating partially parallel acquisitions (GRAPPA)," *Magn. Reson. Med*, vol. 47, no. 6, pp. 1202–1210, 2002. DOI: 10.1002/mrm.10171.

[71] J. Haldar, "Low-rank modeling of local k-space neighborhoods (LORAKS) for constrained MRI," *IEEE Trans. Med. Imaging*, vol. 33, no. 3, pp. 668–681, 2013. DOI: 10.1109/TMI.2013.2293974.

[72] F. Schwabl, "Quantenmechanik: Eine Einführung," in Berlin / Heidelberg / New York: Springer, 1992, ch. 9, pp. 173–180.

[73] F. Bloch, "Nuclear Induction," *Phys. Rev.*, vol. 70, no. 7–8, pp. 460–474, 1946. DOI: 10.1103/PhysRev.70.460.

[74] R. Rzedzian, P. Mansfield, M. Doyle, D. Guilfoyle, B. Chapman, R. Coupland, A. Chrispin, and P. Small, "Real-time nuclear magnetic resonance clinical imaging in paediatrics," *The Lancet*, vol. 322, no. 8362, pp. 1281–1282, 1983. DOI: 10.1016/S0140-6736(83)91153-4.

[75] M. K. Stehling, R. Turner, and P. Mansfield, "Echo-planar imaging: Magnetic resonance imaging in a fraction of a second," *Science*, vol. 254, no. 5028, pp. 43–50, 1991. DOI: 10.1126/science.1925560.

[76] C. Ahn, J. Kim, and Z. Cho, "High-speed spiral-scan echo planar NMR imaging-I," *IEEE Trans. Med. Imaging*, vol. 5, no. 1, pp. 2–7, 1986. DOI: 10.1109/TMI.1986.4307732.

[77] A. Sigfridsson, S. Petersson, C.-J. Carlhäll, and T. Ebbers, "Four-dimensional flow MRI using spiral acquisition," *Magn. Reson. Med.*, vol. 68, no. 4, pp. 1065–1073, 2012. DOI: 10.1002/mrm.23297.

[78] Y. Lim, S. G. Lingala, S. S. Narayanan, and K. S. Nayak, "Dynamic off-resonance correction for spiral real-time MRI of speech," *Magn. Reson. Med.*, vol. 81, no. 1, pp. 234–246, 2019. DOI: 10.1002/mrm.27373.

[79] D. C. Noll, J. M. Pauly, C. H. Meyer, D. G. Nishimura, and A. Macovskj, "Deblurring for non-2D Fourier transform magnetic resonance imaging," *Magn. Reson. Med.*, vol. 25, no. 2, pp. 319–333, 1992. DOI: 10.1002/mrm.1910250210.

[80] B. Sutton, D. Noll, and J. Fessler, "Fast, iterative image reconstruction for MRI in the presence of field inhomogeneities," *IEEE Trans. Med. Imaging*, vol. 22, no. 2, pp. 178–188, 2003. DOI: 10.1109/TMI.2002.808360.

[81] T. Knopp, H. Eggers, H. Dahnke, J. Prestin, and J. Sénégas, "Iterative off-resonance and signal decay correction for improved multi-echo imaging in MRI," *IEEE Trans. Med. Imaging*, vol. 28, no. 3, pp. 394–404, 2009. DOI: 10.1109/TMI.2008.2006526.

[82] C. Findeklee, "Array Noise Matching—Generalization, Proof and Analogy to Power Matching," *IEEE Trans. on Antennas Propag.*, vol. 59, no. 2, pp. 452–459, 2011. DOI: 10.1109/TAP.2010.2096183.

[83] M. Buehrer, K. P. Pruessmann, P. Boesiger, and S. Kozerke, "Array compression for MRI with large coil arrays," *Magn. Reson. Med.*, vol. 57, no. 6, pp. 1131–1139, 2007. DOI: 10.1002/mrm.21237.

[84] P. B. Roemer, W. A. Edelstein, C. E. Hayes, S. P. Souza, and O. M. Mueller, "The NMR phased array," *Magn. Reson. Med.*, vol. 16, no. 2, pp. 192–225, 1990. DOI: 10.1002/mrm.1910160203.

[85] C.-M. Tsai and D. G. Nishimura, "Reduced aliasing artifacts using variable-density k-space sampling trajectories," *Magn. Reson. Med.*, vol. 43, no. 3, pp. 452–458, 2000. DOI: 10.1002/(SICI)1522-2594(200003)43:3<452::AID-MRM18>3.0.CO;2-B.

[86] G. Puy, P. Vandergheynst, and Y. Wiaux, "On variable density compressive sampling," *IEEE Signal Process. Lett.*, vol. 18, no. 10, pp. 595–598, 2011. DOI: 10.1109/LSP.2011.2163712.

[87] N. Gdaniec, H. Eggers, P. Börnert, M. Doneva, and A. Mertins, "Robust abdominal imaging with incomplete breath-holds," *Magn Reson. Med.*, vol. 71, no. 5, pp. 1733–1742, 2014. DOI: 10.1002/mrm.24829.

[88] M. Schmitt, A. Potthast, D. E. Sosnovik, J. R. Polimeni, G. C. Wiggins, C. Triantafyllou, and L. L. Wald, "A 128-channel receive-only cardiac coil for highly accelerated cardiac MRI at 3 Tesla," *Magn. Reson. Med.*, vol. 59, no. 6, pp. 1431–1439, 2008. DOI: 10.1002/mrm.21598.

[89] B. Zhu, J. Z. Liu, S. F. Cauley, B. R. Rosen, and M. S. Rosen, "Image reconstruction by domain-transform manifold learning," *Nature*, vol. 555, no. 7697, pp. 487–492, 2018. DOI: 10.1038/nature25988.

[90] D. Lee, J. Yoo, S. Tak, and J. C. Ye, "Deep residual learning for accelerated MRI using magnitude and phase networks," *IEEE. Trans. Biomed. Eng.*, vol. 65, no. 9, pp. 1985–1995, 2018. DOI: 10.1109/TBME.2018.2821699.

[91] M. Zhang, M. Li, J. Zhou, Y. Zhu, S. Wang, D. Liang, Y. Chen, and Q. Liu, "High-dimensional embedding network derived prior for compressive sensing MRI reconstruction," *Med. Image Anal.*, vol. 64, 101717, 2020. DOI: 10.1016/j.media.2020.101717.

[92] E. U. Saritas, P. W. Goodwill, L. R. Croft, J. J. Konkle, K. Lu, B. Zheng, and S. M. Conolly, "Magnetic particle imaging (MPI) for NMR and MRI researchers," *J. Magn. Reson.*, vol. 229, no. 1, pp. 116–126, 2013. DOI: 10.1016/j.jmr.2012.11.029.

[93] J. Czernin and M. E. Phelps, "Positron emission tomography scanning: Current and future applications," *Annu. Rev. Med.*, vol. 53, no. 1, pp. 89–112, 2002. DOI: 10.1146/annurev.med.53.082901.104028.

[94] G. A. Beller and S. R. Bergmann, "Myocardial perfusion imaging agents: SPECT and PET," *J. Nucl. Cardiol.*, vol. 11, no. 1, pp. 71–86, 2004. DOI: 10.1016/j.nuclcard.2003.12.002.

[95] P. Ludewig, M. Graeser, N. D. Forkert, F. Thieben, J. Rández-Garbayo, J. Rieckhoff, K. Lessmann, F. Förger, P. Szwargulski, T. Magnus, and T. Knopp, "Magnetic particle imaging for assessment of cerebral perfusion and ischemia," *Wiley Interdiscip. Rev.: Nanomed. Nanobiotechnology*, vol. 14, no. 1, e1757, 2022. DOI: 10.1002/wnan.1757.

[96] S. Herz, P. Vogel, T. Kampf, P. Dietrich, S. Veldhoen, M. A. Rückert, R. Kickuth, V. C. Behr, and T. A. Bley, "Magnetic particle imaging–guided stenting," *J. Endovasc. Ther.*, vol. 26, no. 4, pp. 512–519, 2019. DOI: 10.1177/1526602819851202.

[97] M. Gräser, F. Thieben, P. Szwargulski, F. Werner, N. Gdaniec, M. Boberg, F. Griese, M. Möddel, P. Ludewig, D. van de Ven, *et al.*, "Human-sized magnetic particle imaging for brain applications," *Nat. Commun.*, vol. 10, no. 1, 1936, 2019. DOI: 10.1038/s41467-019-09704-x.

[98] K. Them, M. G. Kaul, C. Jung, M. Hofmann, T. Mummert, F. Werner, and T. Knopp, "Sensitivity enhancement in magnetic particle imaging by background subtraction," *IEEE Trans. Med. Imaging*, vol. 35, no. 3, pp. 893–900, 2015. DOI: 10.1109/TMI.2015.2501462.

[99] T. Knopp, N. Gdaniec, R. Rehr, M. Gräser, and T. Gerkmann, "Correction of linear system drifts in magnetic particle imaging," *Phys. Med. Biol.*, vol. 64, no. 12, 125013, 2019. DOI: 10.1088/1361-6560/ab2480.

[100] T. Knopp, *Effiziente Rekonstruktion und Alternative Spulentopologie für Magnetic-Particle-Imaging*. Wiesbaden: Vieweg+Teuber, 2011.

[101] Y. Xiao and J. Du, "Superparamagnetic nanoparticles for biomedical applications," *J. Mater. Chem. B*, vol. 8, no. 3, pp. 354–367, 2020. DOI: 10.1039/C9TB01955C.

[102] S. Ota and Y. Takemura, "Characterization of Néel and Brownian relaxations isolated from complex dynamics influenced by dipole interactions in magnetic nanoparticles," *J. Phys. Chem. C*, vol. 123, no. 47, pp. 28 859–28 866, 2019. DOI: 10.1021/acs.jpcc.9b06790.

[103] S. Chikazumi and C. D. Graham, *Physics of Ferromagnetism* (International Series of Monographs on Physics 94). Oxford: Oxford University Press, 2009.

[104] J. Weizenecker, "The Fokker–Planck equation for coupled Brown–Néel-rotation," *Phys. Med. Biol.*, vol. 63, no. 3, 035004, 2018. DOI: 10.1088/1361-6560/aaa186.

[105] M. Graeser, K. Bente, and T. M. Buzug, "Dynamic single-domain particle model for magnetite particles with combined crystalline and shape anisotropy," *J. Phys. D: Appl. Phys.*, vol. 48, no. 27, 275001, 2015. DOI: 10.1088/0022-3727/48/27/275001.

[106] T. Kluth, P. Szwargulski, and T. Knopp, "Towards accurate modeling of the multidimensional magnetic particle imaging physics," *New J. Phys.*, vol. 21, no. 10, 103032, 2019. DOI: 10.1088/1367-2630/ab4938.

[107] M. Graeser, T. Knopp, P. Szwargulski, T. Friedrich, A. von Gladiss, M. Kaul, K. M. Krishnan, H. Ittrich, G. Adam, and T. M. Buzug, "Towards picogram detection of superparamagnetic iron-oxide particles using a gradiometric receive coil," *Sci. Rep.*, vol. 7, no. 1, 6872, 2017. DOI: 10.1038/s41598-017-06992-5.

[108] O. Kosch, U. Heinen, L. Trahms, and F. Wiekhorst, "Preparing system functions for quantitative MPI," *Int. J. Magn. Part. Imaging*, vol. 3, no. 2, 706002, 2017. DOI: 10.18416/ijmpi.2017.1706002.

[109] A. M. Rauwerdink and J. B. Weaver, "Viscous effects on nanoparticle magnetization harmonics," *J. Magn. Magn. Mater*, vol. 322, no. 6, pp. 609–613, 2010. DOI: 10.1016/j.jmmm.2009.10.024.

[110] A. M. Rauwerdink, E. W. Hansen, and J. B. Weaver, "Nanoparticle temperature estimation in combined ac and dc magnetic fields," *Phys. Med. Biol.*, vol. 54, no. 19, L51, 2009. DOI: 10.1088/0031-9155/54/19/L01.

[111] J. Weizenecker, B. Gleich, J. Rahmer, and J. Borgert, "Micro-magnetic simulation study on the magnetic particle imaging performance of anisotropic mono-domain particles," *Phys. Med. Biol.*, vol. 57, no. 22, 7317, 2012. DOI: 10.1088/0031-9155/57/22/7317.

[112] E. U. Saritas, P. W. Goodwill, G. Z. Zhang, and S. M. Conolly, "Magnetostimulation Limits in Magnetic Particle Imaging," *IEEE Trans. Med. Imaging*, vol. 32, no. 9, pp. 1600–1610, 2013. DOI: 10.1109/TMI.2013.2260764.

[113] T. Knopp, P. Szwargulski, F. Griese, and M. Gräser, "OpenMPIData: An initiative for freely accessible magnetic particle imaging data," *Data Br.*, vol. 28, no. 1, 104971, 2020. DOI: 10.1016/j.dib.2019.104971.

[114] Z. W. Tay, D. W. Hensley, E. C. Vreeland, B. Zheng, and S. M. Conolly, "The relaxation wall: experimental limits to improving MPI spatial resolution by increasing nanoparticle core size," *Biomed. Phys. Eng. Express*, vol. 3, no. 3, 035003, 2017. DOI: 10.1088/2057-1976/aa6ab6.

[115] J. Rahmer, J. Weizenecker, B. Gleich, and J. Borgert, "Signal encoding in magnetic particle imaging: Properties of the system function," *BMC Med. Imaging*, vol. 9, no. 1, 4, 2009. DOI: 10.1186/1471-2342-9-4/FIGURES/12.

[116] K. J. Button, "Microwave ferrite devices: The first ten years," *IEEE Trans. Microw. Theory*, vol. 32, no. 9, pp. 1088–1096, 1984. DOI: 10.1109/TMTT.1984.1132820.

[117] M. Lany, G. Boero, and R. S. Popovic, "Superparamagnetic microbead inductive detector," *Rev. Sci. Instrum.*, vol. 76, no. 8, 084301, 2005. DOI: 10.1063/1.1988131.

[118] M. Maass and A. Mertins, "On the representation of magnetic particle imaging in fourier space," *Int. J. Magn. Part. Imaging*, vol. 6, no. 1, 912001, 2020. DOI: 10.18416/ijmpi.2019.1912001.

[119] S. Ilbey, C. Bariş, B. Top, A. Güngör, T. Çukur, U. Saritas, H. E. Güven,) C. B. Top, H. E. Güven, and T. Çukur, "Fast system calibration with coded calibration scenes for magnetic particle imaging," *IEEE Trans. Med. Imaging*, vol. 38, no. 9, pp. 2070–2080, 2019. DOI: 10.1109/TMI.2019.2896289.

[120] P. W. Goodwill and S. M. Conolly, "Multidimensional x-space magnetic particle imaging," *IEEE Trans. Med. Imaging*, vol. 30, no. 9, pp. 1581–1590, 2011. DOI: 10.1109/TMI.2011.2125982.

[121] T. Knopp, J. Rahmer, T. F. Sattel, S. Biederer, J. Weizenecker, B. Gleich, J. Borgert, and T. M. Buzug, "Weighted iterative reconstruction for magnetic particle imaging," *Phys. Med. Biol.*, vol. 55, no. 6, pp. 1577–1589, 2010. DOI: 10.1088/0031-9155/55/6/003.

[122] S. Kaczmarz, "Angenäherte Auflösung von systemen linearer Gleichungen," *Bull. Int. Acad. Polon. Sci.*, vol. A35, pp. 355–357, 1937.

[123] M. Storath, C. Brandt, M. Hofmann, T. Knopp, J. Salamon, A. Weber, and A. Weinmann, "Edge preserving and noise reducing reconstruction for magnetic particle imaging," *IEEE Trans. Med. Imaging*, vol. 36, no. 1, pp. 74–85, 2016. DOI: 10.1109/TMI.2016.2593954.

[124] F. Lieb and T. Knopp, "A sparse row-action algorithm for Magnetic Particle Imaging," in *Int. J. Magn. Part. Imaging*, vol. 6, 2020, 2009002. DOI: 10.18416/IJMPI.2020.2009002.

[125] L. Nawwas, M. Möddel, T. Knopp, and C. Brandt, "Bias-reduction for sparsity promoting regularization in magnetic particle imaging," in *Int. J. Magn. Part. Imaging*, vol. 6, 2020, 2009041. DOI: 10.18416/IJMPI.2020.2009041.

[126] J. Rahmer, A. Halkola, B. Gleich, I. Schmale, and J. Borgert, "First experimental evidence of the feasibility of multi-color magnetic particle imaging," *Phys. Med. Biol.*, vol. 60, no. 5, 1775, 2015. DOI: 10.1088/0031-9155/60/5/1775.

[127] M. Möddel, C. Meins, J. Dieckhoff, and T. Knopp, "Viscosity quantification using multi-contrast magnetic particle imaging," *New J. Physics*, vol. 20, no. 8, 083001, 2018. DOI: 10.1088/1367-2630/aad44b.

[128] M. Utkur and E. U. Saritas, "Simultaneous temperature and viscosity estimation capability via magnetic nanoparticle relaxation," *Med. Phys.*, vol. 49, no. 4, pp. 2590–2601, 2022. DOI: 10.1002/mp.15509.

[129] M. Mani, M. Jacob, D. Kelley, and V. Magnotta, "Multi-shot sensitivity-encoded diffusion data recovery using structured low-rank matrix completion (MUSSELS)," *Magn. Reson. Med.*, vol. 78, no. 2, pp. 494–507, 2017. DOI: 10.1002/MRM.26382.

[130] F. Ong and M. Lustig, "SigPy: A Python Package for High Performance Iterative Reconstruction," in *Proc. Int. Soc. Magn. Reson. Med.*, vol. 27, 2019, 4819, p. 4819.

[131] Y. Shen, C. Hu, P. Zhang, J. Tian, and H. Hui, "A novel software framework for magnetic particle imaging reconstruction," *Int. J. Imaging Syst. Technol.*, vol. 32, no. 4, pp. 1119–1132, 2022. DOI: 10.1002/ima.22707.

[132] C. Lattner and V. Adve, "LLVM: a compilation framework for lifelong program analysis & transformation," in *International Symposium on Code Generation and Optimization (CGO)*, 2004, pp. 75–86. DOI: 10.1109/CGO.2004.1281665.

[133] N. Parikh and S. Boyd, "Proximal algorithms," *Found. Trends Optim.*, vol. 1, no. 3, pp. 127–239, 2014. DOI: 10.1561/2400000003.

[134] S. J. Inati, J. D. Naegele, N. R. Zwart, V. Roopchansingh, M. J. Lizak, D. C. Hansen, C. Y. Liu, D. Atkinson, P. Kellman, S. Kozerke, *et al.*, "ISMRM Raw data format: A proposed standard for MRI raw datasets," *Magn. Reson. Med.*, vol. 77, no. 1, pp. 411–421, 2017. DOI: 10.1002/MRM.26089.

[135] M. Uecker, P. Lai, M. J. Murphy, P. Virtue, M. Elad, J. M. Pauly, S. S. Vasanawala, and M. Lustig, "ESPIRiT-an eigenvalue approach to autocalibrating parallel MRI: Where SENSE meets GRAPPA," *Magn. Reson. Med.*, vol. 71, no. 3, pp. 990–1001, 2014. DOI: 10.1002/mrm.24751.

[136] L. Dagum and R. Menon, "OpenMP: an industry standard API for shared-memory programming," *IEEE Comput. Sci. Eng.*, vol. 5, no. 1, pp. 46–55, 1998. DOI: 10.1109/99.660313.

[137] R. D. Blumofe, C. F. Joerg, B. C. Kuszmaul, C. E. Leiserson, K. H. Randall, and Y. Zhou, "Cilk: An Efficient Multithreaded Runtime System," *J. Parallel Distrib. Comput.*, vol. 37, no. 1, pp. 55–69, 1996. DOI: 10.1006/JPDC.1996.0107.

[138] A. Kukanov and M. Voss, "The Foundations for Scalable Multi-core Software in Intel Threading Building Blocks," *Intel Technol. J.*, vol. 11, no. 4, pp. 309–322, 2007. DOI: 10.1535/itj.1104.05.

[139] T. Knopp, M. Möddel, F. Griese, F. Werner, P. Szwargulski, N. Gdaniec, and M. Boberg, "MPIFiles. jl: a Julia package for magnetic particle imaging files," *J. Open Source Softw.*, vol. 4, no. 38, 01331, 2019. DOI: 10.21105/joss.01331.

[140] M. Frigo and S. G. Johnson, "FFTW: An adaptive software architecture for the FFT," in *IEEE Int. Conf. Acoust. Speech Signal Process.*, vol. 3, 1998, pp. 1381–1384. DOI: 10.1109/ICASSP.1998.681704.

[141] O. Maier, S. H. Baete, A. Fyrdahl, K. Hammernik, S. Harrevelt, L. Kasper, A. Karakuzu, F. Knoll, and M. Hsieh, "CG-SENSE revisited: Results from the first ISMRM reproducibility challenge," *Magn. Reson. Med.*, vol. 85, no. 4, pp. 1821–1839, 2021. DOI: 10.1002/mrm.28569.

[142] H. Eggers, P. Boernert, and P. Boesiger, "Comparison of gridding-and convolution-based iterative reconstruction algorithms for sensitivity-encoded non-Cartesian acquisitions," in *Proc. Int. Soc. Magn. Reson. Med.*, vol. 10, 2002, 743.

[143] V. Churavy *et al.* "Implement support for object caching through pkgimages." (2023), [Online]. Available: https://github.com/JuliaLang/julia/pull/47184 (visited on 01/12/2023).

[144] K. T. Block and R. Wiggins. "Yarra - A Toolbox for Clinical MRI Research." (), [Online]. Available: http://https://yarra-framework.com.

[145] E. A. Borisch, R. C. Grimm, and S. J. Riederer, "Automated reconstruction processing," in *Proc. Intl. Soc. Mag. Reson. Med*, vol. 27, 2019, 4838.

[146] L. Kasper, M. Engel, J. Heinzle, M. Mueller-Schrader, N. N. Graedel, J. Reber, T. Schmid, C. Barmet, B. J. Wilm, K. E. Stephan, and K. P. Pruessmann, "Advances in spiral fMRI: A high-resolution study with single-shot acquisition," *Neuroimage*, vol. 246, 118738, 2022. DOI: 10.1016/J.NEUROIMAGE.2021.118738.

[147] S. Ramani and J. A. Fessler, "Parallel MR image reconstruction using augmented Lagrangian methods," *IEEE Trans. Med. Imaging*, vol. 30, no. 3, pp. 694–706, 2010. DOI: 10.1109/TMI.2010.2093536.

[148] D. S. Weller, S. Ramani, and J. A. Fessler, "Augmented Lagrangian with Variable Splitting for Faster Non-Cartesian L_1-SPIRiT MR Image Reconstruction," *IEEE Trans. Med. Imaging*, vol. 33, no. 2, pp. 351–361, 2013. DOI: 10.1109/TMI.2013.2285046.

[149] L. C. Man, J. M. Pauly, and A. Macovski, "Multifrequency interpolation for fast off-resonance correction," *Magn. Reson. Med.*, vol. 37, no. 5, pp. 785–792, 1997. DOI: 10.1002/MRM.1910370523.

[150] N. Halko, P. G. Martinsson, and J. A. Tropp, "Finding Structure with Randomness: Probabilistic Algorithms for Constructing Approximate Matrix Decompositions," *SIAM Rev.*, vol. 53, no. 2, pp. 217–288, 2011. DOI: 10.1137/090771806.

[151] T. Pock and A. Chambolle, "Diagonal preconditioning for first order primal-dual algorithms in convex optimization," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2011, pp. 1762–1769. DOI: 10.1109/ICCV.2011.6126441.

[152] Y. Wang and T. Liu, "Quantitative susceptibility mapping (QSM): decoding MRI data for a tissue magnetic biomarker," *Magn. Reson. Med.*, vol. 73, no. 1, pp. 82–101, 2015. DOI: 10.1002/mrm.25358.

[153] "MEDI toolbox." ().

[154] D. H. Kim, E. Adalsteinsson, and D. M. Spielman, "Simple analytic variable density spiral design," *Magn. Reson. Med.*, vol. 50, no. 1, pp. 214–219, 2003. DOI: 10.1002/MRM.10493.

[155] S. G. Lingala, Y. Hu, E. DiBella, and M. Jacob, "Accelerated dynamic MRI exploiting sparsity and low-rank structure: kt SLR," *IEEE Trans. Med. Imaging*, vol. 30, no. 5, pp. 1042–1054, 2011. DOI: 10.1109/TMI.2010.2100850.

[156] L. De Lathauwer, B. De Moor, and J. Vandewalle, "A Multilinear Singular Value Decomposition," *SIAM J. Matrix Anal. Appl.*, vol. 21, no. 4, pp. 1253–1278, 2006. DOI: 10.1137/S0895479896305696.

[157] M. E. Kilmer, K. Braman, N. Hao, and R. C. Hoover, "Third-Order Tensors as Operators on Matrices: A Theoretical and Computational Framework with Applications in Imaging," *SIAM J. Matrix Anal. Appl.*, vol. 34, no. 1, pp. 148–172, 2013. DOI: 10.1137/110837711.

[158] I. V. Oseledets, "Tensor-Train Decomposition," *SIAM J. Sci. Comput.*, vol. 33, no. 5, pp. 2295–2317, 2011. DOI: 10.1137/090752286.

[159] G. Bergqvist and E. Larsson, "The higher-order singular value decomposition: Theory and an application," *IEEE Signal Process. Mag.*, vol. 27, no. 3, pp. 151–154, 2010. DOI: 10.1109/MSP.2010.936030.

[160] O. Semerci, N. Hao, M. E. Kilmer, and E. L. Miller, "Tensor-Based Formulation and Nuclear Norm Regularization for Multienergy Computed Tomography," *IEEE Trans. Image Process.*, vol. 23, no. 4, pp. 1678–1693, 2014. DOI: 10.1109/TIP.2014.2305840.

[161] F. Lieb and H.-G. Stark, "Background removal by mixing factor based filtering of the system matrix," in *Int. Workshop Magn. Part. Imaging Book Abstr.*, vol. 9, 2019, pp. 33–34.

[162] A. Rajwade, A. Rangarajan, and A. Banerjee, "Image denoising using the higher order singular value decomposition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 4, pp. 849–862, 2013. DOI: 10.1109/TPAMI.2012.140.

[163] M. Signoretto, L. De Lathauwer, and J. A. K. Suykens, "Nuclear norms for tensors and their use for convex multilinear estimation," *Submitted Linear Algebra Appl.*, 2010.

[164] D. L. Donoho, "De-Noising by Soft-Thresholding," *IEEE Trans. Inf. Theory*, vol. 41, no. 3, pp. 613–627, 1995. DOI: 10.1109/18.382009.

[165] J. F. Cai, E. J. Candès, and Z. Shen, "A Singular Value Thresholding Algorithm for Matrix Completion," *SIAM J. Optim.*, vol. 20, no. 4, pp. 1956–1982, 2010. DOI: 10.1137/080738970.

[166] A. Weber, F. Werner, J. Weizenecker, T. M. Buzug, and T. Knopp, "Artifact free reconstruction with the system matrix approach by overscanning the field-free-point trajectory in magnetic particle imaging," *Phys. Med. Biol.*, vol. 61, no. 2, 475, 2015. DOI: 10.1088/0031-9155/61/2/475.

[167] Z. Zhang and S. Aeron, "Exact Tensor Completion Using t-SVD," *IEEE Trans. Signal Process.*, vol. 65, no. 6, pp. 1511–1526, 2017. DOI: 10.1109/TSP.2016.2639466.

[168] J. A. Bengua, H. N. Phien, H. D. Tuan, and M. N. Do, "Efficient Tensor Completion for Color Image and Video Recovery: Low-Rank Tensor Train," *IEEE Trans. Image Process.*, vol. 26, no. 5, pp. 2466–2479, 2017. DOI: 10.1109/TIP.2017.2672439.

[169] S. Joshi and S. Boyd, "Sensor selection via convex optimization," *IEEE Trans. Signal Process.*, vol. 57, no. 2, pp. 451–462, 2009. DOI: 10.1109/TSP.2008.2007095.

[170] S. Liu, S. P. Chepuri, M. Fardad, E. Masazade, G. Leus, and P. K. Varshney, "Sensor selection for estimation with correlated measurement noise," *IEEE Trans. Signal Process.*, vol. 64, no. 13, pp. 3509–3522, 2016. DOI: 10.1109/TSP.2016.2550005.

[171] J. Swärd, F. Elvander, and A. Jakobsson, "Designing sampling schemes for multi-dimensional data," *Signal Process.*, vol. 150, no. 1, pp. 1–10, 2018. DOI: 10.1016/j.sigpro.2018.03.011.

[172] Z. Ben-Haim and Y. C. Eldar, "The Cramér-Rao Bound for Estimating a Sparse Parameter Vector," *IEEE Trans. Signal Process.*, vol. 58, no. 6, pp. 3384–3389, 2010. DOI: 10.1109/TSP.2010.2045423.

[173] S. J. Reeves and Z. Zhe, "Sequential algorithms for observation selection," *IEEE Trans. Signal Process.*, vol. 47, no. 1, pp. 123–132, 1999. DOI: 10.1109/78.738245.

[174] R. Broughton, I. Coope, P. Renaud, and R. Tappenden, "Determinant and exchange algorithms for observation subset selection," *IEEE Trans. Image Process.*, vol. 19, no. 9, pp. 2437–2443, 2010. DOI: 10.1109/TIP.2010.2048150.

[175] W. W. Hager, "Updating the inverse of a matrix," *SIAM Rev.*, vol. 31, no. 2, pp. 221–239, 1989. DOI: 10.1137/1031049.

[176] 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.*, vol. 3, no. 1, pp. 1–122, 2010. DOI: 10.1561/2200000016.

[177] M. Boberg, T. Knopp, P. Szwargulski, and M. Möddel, "Generalized MPI Multi-Patch Reconstruction Using Clusters of Similar System Matrices," *IEEE Trans. Med. Imaging*, vol. 39, no. 5, pp. 1347–1358, 2020. DOI: 10.1109/TMI.2019.2949171.

[178] T. B. Smith and K. S. Nayak, "Automatic off-resonance correction in spiral imaging with piecewise linear autofocus," *Magnetic Resonance in Medicine*, vol. 69, no. 1, pp. 82–90, 2013. DOI: https://doi.org/10.1002/mrm.24230.

[179] G. Wang, T. Luo, J.-F. Nielsen, D. C. Noll, and J. A. Fessler, "B-Spline Parameterized Joint Optimization of Reconstruction and K-Space Trajectories (BJORK) for Accelerated 2D MRI," *IEEE Trans. Med. Imaging*, vol. 41, no. 9, pp. 2318–2330, 2022. DOI: 10.1109/TMI.2022.3161875.

[180] J. W. Cooley and J. W. Tukey, "An algorithm for the machine calculation of complex Fourier series," *Math. Comput.*, vol. 19, no. 90, pp. 297–301, 1965. DOI: 10.2307/2003354.

[181] H. Eggers, T. Knopp, and D. Potts, "Field inhomogeneity correction based on gridding reconstruction for magnetic resonance imaging," *IEEE Trans. Med. Imaging*, vol. 26, no. 3, pp. 374–384, 2007. DOI: 10.1109/TMI.2006.891502.

[182] S. Boyd and L. Vandenberghe, "Convex optimization," in Cambridge: Cambridge university press, 2004, ch. 3, pp. 90–95.

[183] L. M. Bregman, "The relaxation method of finding the common point of convex sets and its application to the solution of problems in convex programming," *USSR Comput. Math. Math. Phys.*, vol. 7, no. 3, pp. 200–217, 1967. DOI: 10.1016/0041-5553(67)90040-7.

[184] W. Yin, S. Osher, D. Goldfarb, and J. Darbon, "Bregman iterative algorithms for ℓ_1-minimisation with applications to compressed sensing," *SIAM J. Imaging Sci*, vol. 1, no. 1, pp. 143–168, 2008. DOI: 10.1137/070703983.

A Appendix

A.1 Efficient Evaluation of Fourier-type Operators

A central part of MRI image reconstruction is concerned with the efficient evaluation of the signal encoding operators $\mathbf{E}$. As outlined in chapter 4, a direct storage and evaluation of the corresponding matrix representation is usually not feasible due to the large size of the underlying images and the correspondingly large matrix sizes. For this reason, this section summarizes the most popular approaches for evaluating MRI imaging operators. As outlined in chapter 4, the MRI imaging operator for an L-channel acquisition can be decomposed as

$$\mathbf{S} = (\mathbf{F}\mathbf{P}_1 \dots \mathbf{F}\mathbf{P}_L)^{\mathsf{T}},$$

where the operators $\mathbf{P}_l$ denote the multiplication with the given coil sensitivity maps and the operator $\mathbf{F}$ is a Fourier-type operator defined in Eqs. (4.25) and (4.26). Since the operators $\mathbf{P}_l$ are diagonal they can be evaluated in a straight-forward manner by element-wise multiplication. In the following we therefore focus on the evaluation of the Fourier-type operator $\mathbf{F}$.

To simplify the treatment, we assume the imaging grid points to lie on a D-dimensional, rectangular grid of the form

$$\Upsilon_{\mathbf{N}} = \left\{-\frac{N_1}{2}, \dots, \frac{N_1}{2} - 1\right\} \times \dots \times \left\{-\frac{N_D}{2}, \dots, \frac{N_D}{2} - 1\right\},$$

where we used the notation $\mathbf{N} = (N_1 \ \dots \ N_D)^{\mathsf{T}}$ to collect the number of grid points along each dimension. Note that this representation of the imaging grid is very generic and can easily be achieved by a proper choice of units. If we now assume a discrete image and compute the DFT for a k-space node $\mathbf{k}$, we find that the result is periodic in each component of $\mathbf{k}$ with a respective period of 1. For this reason, we also constrain the space of k-space nodes so that $\mathbf{k} \in \mathbb{T}^D = [-1/2, 1/2) \times \dots \times [-1/2, 1/2)$. For latter reference, we also introduce the Cartesian k-space grid

$$\tilde{\Upsilon}_{\mathbf{N}} = \left\{-\frac{1}{2}, \dots, \frac{1}{2} - \frac{1}{N_1}\right\} \times \dots \times \left\{-\frac{1}{2}, \dots, \frac{1}{2} - \frac{1}{N_D}\right\},$$

which is the minimum Cartesian grid required to achieve Nyquist sampling for the assumed image.

A.1.1 Homogeneous B_0-field

For most MRI acquisitions, one makes the assumption of a homogeneous B_0-field. In this case, the operator $\mathbf{F}$ takes the form

$$\mathbf{F} = \left(\frac{1}{\sqrt{N}} \mathrm{e}^{-2\pi \mathrm{i} \mathbf{k}_j \cdot \mathbf{r}_i} \right)_{j=1,\ldots,J; i=1,\ldots,N} \tag{A.1}$$

and thus reduces to a simple DFT. A further simplification arises if the sampling trajectory has equidistant k-space samples, as is the case for the popularly used Cartesian trajectories. In this case, the operator $\mathbf{F}$ can be evaluated efficiently using the well known FFT [3, 140, 180]. Use of the FFT leads to a significant reduction in computational effort required for evaluating $\mathbf{F}$. While a naive multiplication with the matrix $\mathbf{F}$ has a computational complexity of $\mathcal{O}(JN)$, the FFT has a complexity of $\mathcal{O}(N \log N)$, which quickly amortizes for moderate to large problem sizes.

When turning to the case of non-Cartesian trajectories, things become more complicated, because the FFT cannot directly be applied. Nevertheless, $\mathbf{F}$ can be evaluated efficiently using the NFFT, for which short derivation is provided in the following. A more detailed treatment of the NFFT can be found in [J7].

The basic idea underlying the NFFT is to interpolate the non-Cartesian k-space nodes onto a larger Cartesian grid, so that the FFT can be applied to the interpolated data. To formalize this, we consider a 1d signal $\mathbf{f} \in \mathbb{C}^N$. To interpolate the corresponding Fourier space data, we use a window function $\tilde{\varphi} \in L^2(\mathbb{R}^D) \cap L^1(\mathbb{R}^D) \cap BV(\mathbb{R}^D)$, where the function space BV encompasses all functions with a bounded total variation. Moreover, we assume the window function to be even and well localized in both spatial and frequency domain. To derive the NFFT, let us consider its inverse Fourier transform

$$\varphi(\mathbf{r}_i) = \int_{\mathbb{R}^D} \tilde{\varphi}(\mathbf{k}) \mathrm{e}^{2\pi \mathrm{i} \mathbf{k} \cdot \mathbf{r}_i} \, \mathrm{d}^D k = \sum_{\boldsymbol{\kappa} \in \mathbb{Z}^D} \int_{\mathbb{T}^D} \tilde{\varphi}(\mathbf{k} + \boldsymbol{\kappa}) \mathrm{e}^{2\pi \mathrm{i} (\mathbf{k} + \boldsymbol{\kappa}) \cdot \mathbf{r}_i} \, \mathrm{d}^D k.$$

For the assumed imaging points, we know that the components of $\mathbf{r}_i$ only take integer values. In this case, we can make use of the periodicity of the complex exponential. If we further introduce the one-periodization of the window function $\tilde{\varphi}(\mathbf{k}) = \sum_{l \in \mathbb{Z}^3} \hat{\varphi}(\mathbf{k} + \mathbf{l})$, we can rewrite the inverse Fourier transform as

$$\varphi(\mathbf{r}_i) = \int_{\mathbb{T}^D} \tilde{\varphi}(\mathbf{k}) \mathrm{e}^{2\pi \mathrm{i} \mathbf{k} \cdot \mathbf{r}_i} \, \mathrm{d}^D k = \int_{\mathbb{T}^D} \tilde{\varphi}(\mathbf{k} - \mathbf{k}') \mathrm{e}^{2\pi \mathrm{i} (\mathbf{k} - \mathbf{k}') \cdot \mathbf{r}_i} \, \mathrm{d}^D k'.$$

Here expression on the right-hand side was obtained by applying the substitution $\mathbf{k} \to \mathbf{k} - \mathbf{k}'$. If we further perform division by $\varphi(\mathbf{r}_i)$ and $\mathrm{e}^{2\pi \mathrm{i} \mathbf{k} \cdot \mathbf{r}_i}$, this leads us to the expression

$$\mathrm{e}^{-2\pi \mathrm{i} \mathbf{k} \cdot \mathbf{r}_i} = \frac{1}{\varphi(\mathbf{r}_i)} \int_{\mathbb{T}^D} \tilde{\varphi}(\mathbf{k} - \mathbf{k}') \mathrm{e}^{-2\pi \mathrm{i} \mathbf{k}' \cdot \mathbf{r}_i} \, \mathrm{d}^D k'$$

Next, we introduce an oversampled k-space grid with nodes $\mathbf{l} \in \tilde{\Upsilon}_{\alpha\mathbf{N}}$ and use a Riemann sum to approximate the integral, which yields

$$\mathrm{e}^{-2\pi \mathrm{i}\mathbf{k}\cdot\mathbf{r}_i} \approx \frac{1}{\alpha^D N \varphi(\mathbf{r}_i)} \sum_{\mathbf{l}\in\tilde{\Upsilon}_{\alpha\mathbf{N}}} \tilde{\varphi}(\mathbf{k}-\mathbf{l})\mathrm{e}^{-2\pi \mathrm{i}\mathbf{l}\cdot\mathbf{r}_i} \tag{A.2}$$

This approximation is useful, because it relates the exponential for an arbitrary k-space node to that for a set of Cartesian k-space nodes. This allows us to use FFTs for the computation of the NDFT. As a matter of fact, performing the sum over $\mathbf{l}$ still requires $\mathcal{O}(\alpha^D N^2)$ operations. To circumvent this issue the interpolation kernel is typically chosen such that only a few function values centered around 0 are significant. Based on this, we introduce the truncated kernel

$$\tilde{\psi}(\kappa) := \begin{cases} \tilde{\varphi}(\mathbf{k}) & \text{if } \mathbf{k} \in [-\frac{m}{\alpha N_1}, \frac{m}{\alpha N_1}] \times \cdots \times [-\frac{m}{\alpha N_D}, \frac{m}{\alpha N_D}] \\ 0 & \text{else} \end{cases},$$

which results in the approximation

$$\mathrm{e}^{-2\pi \mathrm{i}\mathbf{k}\cdot\mathbf{r}_i} \approx \frac{1}{\alpha^D N \varphi(\mathbf{r}_i)} \sum_{\mathbf{l}\in\tilde{\Upsilon}_{\alpha\mathbf{N}}} \tilde{\psi}(\mathbf{k}-\mathbf{l})\mathrm{e}^{-2\pi \mathrm{i}\mathbf{l}\cdot\mathbf{r}_i}. \tag{A.3}$$

Note that in this expression the sum over l contains at most $2m+1$ non-zero elements, which means that it can be computed quite efficiently. To obtain the final expression evaluated by the NFFT, we plug (A.3) into the definition of the DFT, which yields

$$\begin{aligned} s_j &= \frac{1}{\sqrt{N}} \sum_{i=1}^{N} f_i \mathrm{e}^{-2\pi \mathrm{i}\mathbf{k}_j\cdot\mathbf{r}_i} \\ &\approx \sum_{\mathbf{l}\in\tilde{\Upsilon}_{\alpha\mathbf{N}}} \tilde{\psi}(\mathbf{k}_j-\mathbf{l}) \sum_{i=1}^{N} \frac{f_i}{\alpha^D N^{3/2} \varphi(\mathbf{r}_i)} \mathrm{e}^{-2\pi \mathrm{i}\mathbf{l}\cdot\mathbf{r}_i}. \end{aligned} \tag{A.4}$$

Based on this expression, the non-equidistant DFT can efficiently by evaluated as follows:

1. *apodization*: compute the index-wise products $\frac{f_i}{\alpha^D N^{3/2}\varphi(\mathbf{r}_i)}$ in $\mathcal{O}(N)$.
2. *FFT*: compute the sum over i using an oversampled FFT with complexity $\mathcal{O}(\alpha^D N \log(\alpha^D N))$.
3. *convolution*: compute the sum over l in $\mathcal{O}(m^D J)$.

Using this algorithm, the NFFT thus allows evaluating the operator $\mathbf{F}$ with a complexity of $\mathcal{O}(m^D J + \alpha^D N \log(\alpha^D N))$. Similarly, the adjoint DFT can be computed efficiently by performing the respective adjoint operations.

A.1.2 Inhomogeneous B_0-field

In some common MRI sequences, the assumption of negligible field inhomogeneity starts breaking down. In order to avoid artifacts, one should perform image reconstruction based on the general signal model (4.17). To simplify the treatment, we make the assumption that the effects of T_2^*-relaxation remain constant during acquisition of one profile. In this case, the operator $\mathbf{F}$ takes the form

$$\mathbf{F} = \left(\frac{1}{\sqrt{N}} \mathrm{e}^{-2\pi \mathrm{i} \mathbf{k}_j \cdot \mathbf{r}_i} \mathrm{e}^{-\mathrm{i}\Delta\omega_0(\mathbf{r}_i)(t_j - t_E)} \right)_{j=1,\dots,J; i=1,\dots,N}.$$

A problem with this operator is that the non-zero field-map-term $\mathrm{e}^{-\mathrm{i}\Delta\omega_0(\mathbf{r}_i)(t_j-t_E)}$ destroys the Fourier structure of the operator. Nevertheless, it is desirable to use FFT-based algorithms in order to efficiently evaluate this operator. To achieve this, we aim to approximate the problematic exponential using a low rank matrix

$$\mathbf{C} = \left(\mathrm{e}^{-\mathrm{i}\Delta\omega_0(\mathbf{r}_i)(t_j-t_E)} \right)_{j=1,\dots,J; i=1,\dots,N} \approx \mathbf{A}\mathbf{B}^T = \sum_{\kappa=1}^{K} \mathbf{a}_{:,\kappa} \mathbf{b}_{:,\kappa}^{\mathsf{T}},$$

where $\mathbf{A} \in \mathbb{C}^{J \times K}$ and $\mathbf{B} \in \mathbb{C}^{N \times K}$. When plugging this approximation into our expression for $\mathbf{F}$, the latter can be rearranged to read

$$\mathbf{F} = \sum_{\kappa=1}^{K} \mathrm{diag}\left(\mathbf{a}_{:,\kappa}\right) \mathbf{F}_0 \mathrm{diag}\left(\mathbf{b}_{:,\kappa}\right), \tag{A.5}$$

where $\mathbf{F}_0 = \left(1/\sqrt{N} \mathrm{e}^{-2\pi \mathrm{i} \mathbf{k}_j \cdot \mathbf{r}_i} \right)_{j=1,\dots,J; i=1,\dots,N}$ is the DFT arising when assuming a homogeneous B_0 field. This suggests the following efficient algorithm to apply $\mathbf{F}$ to a given image vector:

1. For $\kappa = 1, \dots, K$ compute the elementwise product with $\mathbf{b}_{:,\kappa}$.
2. Apply the operator $\mathbf{F}_0$ using K FFTs or NFFTs.
3. Compute the elementwise products with the K vectors $\mathbf{a}_{:,\kappa}$ and perform the outer sum over κ.

In this algorithm, the main computational effort arises from the K FFTs or NFFTs in step 2, both of which have a loglinear complexity. In contras, steps 1 and 3 can be performed with respective complexities of $\mathcal{O}(KN)$ and $\mathcal{O}(KJ)$.

The remaining aspect to be addressed is how to determine suitable vectors $\mathbf{a}_{:,\kappa}$ and $\mathbf{b}_{:,\kappa}$. A choice that minimizes the ℓ_2-approximation error of $\mathbf{C}$ can be obtained by setting the $\mathbf{a}_{:,\kappa}$ and $\mathbf{b}_{:,\kappa}$ respectively to the left and right singular vectors of $\mathbf{C}$. As a matter of fact, this approach involves computing the SVD of $\mathbf{C}$. In practice, this is not feasible due to the large size of the latter. For this reason, it is common practice in MRI, to use a heuristic approximation for one set of basis functions and optimize the remaining ones in a data-driven manner.

The time segmentation approach, uses translates in time as basis functions $\mathbf{b}_{:,\kappa}$, i.e. we choose basis functions

$$\mathbf{b}_{:,\kappa} = \left(\mathrm{e}^{-\mathrm{i}\Delta\omega_0(\mathbf{r}_i)\bar{t}_\kappa} \right)_{i=1,\dots,N}, \tag{A.6}$$

for a suitable, typically equispaced, set of times $\bar{t}_\kappa$ that cover the sampling interval used during the measurements [42, 80, 181]. To determine a suitable choice for $\mathbf{A}$ given $\mathbf{B}$, it makes sense to minimize the ℓ_2-approximation error of $\mathbf{C}$. To this end, let us consider the j^{th} row of $\mathbf{A}$. To minimize the ℓ_2-approximation error, the former should be a solution of the minimization problem

$$\underset{\mathbf{a}_{j,:}}{\operatorname{argmin}} \sum_{i=1}^{N} \left| \mathrm{e}^{-\mathrm{i}\Delta\omega_0(\mathbf{r}_i)t_j} - \mathbf{a}_{j,:}^{\mathsf{T}} \mathbf{b}_{i,:} \right|^2 . \tag{A.7}$$

While providing good results, a drawback of this method is the unnecessarily large problem size, which implies long computation times. In practice, the rank K of the approximation is chosen such that $K \ll N$. For this reason the determination of the $\mathbf{a}_{j,:}$ does not require the use of all N available field map samples.

A more efficient alternative is the use of histograms proposed by Fessler *et al.* [42]. Following this approach, we consider a histogram $\{(\Delta\omega_{0,i}, h_i) | i = 1, \ldots, N_{\text{hist}}\}$ of values contained in the field map. Here $\Delta\omega_{0,i}$ denotes the field map value at the center of the i^{th} bin and h_i is the number of field map values in the corresponding bin. Based on this histogram, we construct a reduced version of the the $\mathbf{B}$-matrix

$$\mathbf{B}_{\text{hist}} = \left(\mathrm{e}^{-\mathrm{i}\Delta\omega_{0,i}\bar{t}_\kappa} \right)_{i=1,\ldots,N_{\text{hist}};\kappa=1,\ldots,K} .$$

and determine the $\mathbf{a}_{j,:}$ by minimizing the following histogram-based surrogate of the approximation error

$$\underset{\mathbf{a}_{j,:}}{\operatorname{argmin}} \sum_{i=1}^{N_{\text{hist}}} h_i \left| \mathrm{e}^{-\mathrm{i}\Delta\omega_{0,i}t_j} - \mathbf{a}_{j,:}^{\mathsf{T}} \mathbf{b}_{\text{hist},i,:} \right|^2 . \tag{A.8}$$

Here $\mathbf{b}_{\text{hist},i,:}$ denotes the i^{th} row of $\mathbf{B}_{\text{hist}}$.

A popular alternative to TS is the frequency segmentation approach [42, 149]. In this approach, translates in the frequency domain are used for the $\mathbf{a}_{:,}\kappa$, i.e. one chooses

$$\mathbf{a}_{:,\kappa} = \left(\mathrm{e}^{-\mathrm{i}\overline{\Delta\omega}_{0,\kappa}t_j} \right)_{j=1,\ldots,J} , \tag{A.9}$$

for a suitable set of frequency samples $\bar{t}_\kappa$. In contrast to the time segmentation approach, it is common to use a non-equispaced set of frequency samples. This is needed to take into account the generally non-uniform field map distribution. To determine the corresponding matrix $\mathbf{B}$, we again seek to minimize the ℓ_2-approximation error. Thus, for each row i, the $\mathbf{b}_{i,:}$ can be determined by solving

$$\underset{\mathbf{b}_{i,:}}{\operatorname{argmin}} \sum_{j=1}^{J} \left| \mathrm{e}^{-\mathrm{i}\Delta\omega_0(\mathbf{r}_i)t_j} - \mathbf{a}_{j,:}^{\mathsf{T}} \mathbf{b}_{i,:} \right| . \tag{A.10}$$

A.2 Constrained Optimization Methods

A.2.1 Primal Dual Hybrid Gradient Method

In this section we provide a derivation of the PDHG method, which can be used for the solution of unconstrained regularized least-squares problems [55]. In particular, we use it in chapter 7 to obtain an efficient algorithm for the reconstruction of non-Cartesian MRI data. In the following derivation, we consider a generic optimization problem of the form

$$\min_{\mathbf{x}} \mathcal{D}(\mathbf{S}\mathbf{x}) + \mathcal{R}(\mathbf{x}). \tag{A.11}$$

When choosing $\mathcal{D}(\mathbf{x}) = \frac{1}{2}\left\|\mathbf{y} - \mathbf{x}\right\|_2^2$ as a measure for data fidelity, this formulation encompasses many of the problems considered in this thesis, such as the MRI reconstruction problem (6.1) or the problem of recovering one row of a system matrix using a given set of calibration measurements.

To derive an equivalent primal dual formulation, we introduce the convex conjugate function

$$\mathcal{D}^{\star}(\mathbf{v}) = \max_{\mathbf{x}} <\mathbf{v}, \mathbf{x}> - \mathcal{D}(\mathbf{x}).$$

An important property of it is that the biconvex conjugate of a convex function fulfills $\mathcal{D}^{\star\star} = \mathcal{D}$ [182]. Making use of this property, the primal problem (A.11) can be rewritten as the saddle-point problem

$$\min_{\mathbf{x}} \max_{\mathbf{v}} \mathcal{R}(\mathbf{x}) + <\mathbf{S}\mathbf{x}, \mathbf{v}> - \mathcal{D}^{\star}(\mathbf{v}). \tag{A.12}$$

Instead of minimizing the primal problem, we now aim to solve the primal-dual problem (A.12). To do so, we make use of the principle that the unique minimizer of a convex function f is also a fixed point of the corresponding proximal map [133]. Formally, this means that the minimizer $\mathbf{x}$ of f fulfills the fixed-point condition

$$\mathbf{x} = \text{prox}_f(\mathbf{x}),$$

where the proximal map is defined as

$$\text{prox}_f(\mathbf{x}) = \frac{1}{2}\underset{\mathbf{u}}{\text{argmin}} \ \left\|\mathbf{u} - \mathbf{x}\right\|_2^2 + f(\mathbf{u}). \tag{A.13}$$

Based on this concept, we alternatingly perform proximal minimization with respect to the primal variable $\mathbf{x}$ and proximal maximization with respect to the dual variable $\mathbf{v}$. Performing proximal maximization with respect to $\mathbf{v}$ leads to the update

$$\mathbf{v}^{k+1} = \text{prox}_{\sigma\mathcal{D}^*(\cdot) - \sigma<\mathbf{S}\mathbf{x}^{\mathbf{k}}, \cdot>}(\mathbf{v}^k) = \text{prox}_{\sigma\mathcal{D}^*}(\mathbf{v}^k + \sigma\mathbf{S}\mathbf{x}^k). \tag{A.14}$$

Similarly, the update with respect to $\mathbf{x}$ takes the form

$$\mathbf{x}^{k+1} = \text{prox}_{\tau\mathcal{R}(\cdot)+\tau<\cdot,\mathbf{S}^{\mathsf{H}}\mathbf{v}^{k+1}>}(\mathbf{v}^k) = \text{prox}_{\tau\mathcal{R}}(\mathbf{x}^k - \tau\mathbf{S}^{\mathsf{H}}\mathbf{v}^{k+1}). \tag{A.15}$$

The PDHG method can now be obtained by iterating between these update steps. If we further consider the quadratic data fidelity $\mathcal{D}(\mathbf{x}) = \frac{1}{2}\|\mathbf{y} - \mathbf{x}\|_2^2$, the PDHG thus boils down to the iteration

$$\begin{aligned}
\mathbf{v}^{k+1} &= \text{prox}_{\sigma\mathcal{D}^\star}(\mathbf{v}^k + \sigma\mathbf{S}\bar{\mathbf{x}}^k) = (\mathbf{1}+\sigma)^{-1}(\mathbf{v}^k + \sigma(\mathbf{S}\bar{\mathbf{x}}^k - \mathbf{y})) \\
\mathbf{x}^{k+1} &= \text{prox}_{\tau\mathcal{R}}(\mathbf{x}^k - \tau\mathbf{S}^{\mathsf{H}}\mathbf{v}^{k+1}) \\
\bar{\mathbf{x}}^{k+1} &= \mathbf{x}^{k+1} + \Theta(\mathbf{x}^{k+1} - \mathbf{x}^k)
\end{aligned} . \tag{A.16}$$

The third step in this scheme is an over-relaxation step which is commonly included. The size of this step is controlled by the parameter $\Theta \in [0, 1]$. To guarantee convergence, the primal and dual step-sizes τ and σ should fulfill $\lambda_{\text{max}}(\mathbf{S}\mathbf{S}^{\mathsf{H}})^{-1}$, where λ_{max} denotes the maximum eigenvalue [55]. As a final remark, note that the choice of $\mathcal{D}$ only enters in the $\mathbf{v}$-update, where we made use of the fact that $\text{prox}_{\sigma\mathcal{D}^\star}(\mathbf{v}) = (\mathbf{1}+\sigma)^{-1}(\mathbf{v} - \sigma\mathbf{y})$ for the quadratic data-fidelity term.

An interesting property of the PDHG method is that the dual variable $\mathbf{v}$ resides in the range of the operator $\mathbf{S}$. Thus, convergence can be accelerated by applying a preconditioner $\mathbf{Q}$ in the range of $\mathbf{S}$, as was proposed in [151]. The resulting preconditionned PDHG iteration reads

$$\begin{aligned}
\mathbf{v}^{k+1} &= (\mathbf{1}+\sigma\mathbf{Q})^{-1}(\mathbf{v}^k + \sigma\mathbf{Q}(\mathbf{S}\bar{\mathbf{x}}^k - \mathbf{y})) \\
\mathbf{x}^{k+1} &= \text{prox}_{\tau\mathcal{R}}(\mathbf{x}^k - \tau\mathbf{S}^{\mathsf{H}}\mathbf{v}^{k+1}) \\
\bar{\mathbf{x}}^{k+1} &= \mathbf{x}^{k+1} + \Theta(\mathbf{x}^{k+1} - \mathbf{x}^k)
\end{aligned} . \tag{A.17}$$

with the primal and dual step sizes fulfilling $\sigma\tau < \lambda_{\text{max}}(\mathbf{Q}\mathbf{S}\mathbf{S}^{\mathsf{H}})^{-1}$ to guarantee convergence. The last condition implies that large step sizes and thus fast convergence can be achieved when $\mathbf{Q}\mathbf{S}\mathbf{S}^{\mathsf{H}} \approx \mathbf{1}$, so that the maximum eigenvalue becomes minimal. Thus, it makes sense to search for a preconditioner which provides a good approximation to $(\mathbf{S}\mathbf{S}^{\mathsf{H}})^{-1}$.

Another important property of the PDHG method is that all steps of the iterative scheme have a quite simple form, which makes it easy to apply. In the $\mathbf{v}$-update, the main bottleneck is the inversion of the matrix $(\mathbf{1}+\sigma\mathbf{Q})$. Note however that this inversion becomes trivial when $\mathbf{Q}$ is chosen to be a diagonal matrix, as was done in chapter 7. The $\mathbf{x}$-update takes the form of a proximal map. The cost for computing it strongly depends on the form of the regularization function at hand. However, it can be computed efficiently for many commonly used regularization functions, as was discussed in section 6.3. Finally, the computation of $\bar{\mathbf{x}}$ only involves element-wise operations and thus has linear computational complexity.

A.2.2 Split Bregman Method & ADMM

This section provides an introduction to the Split Bregman method, which is used for the recovery of MPI system matrices in chapter 8. This method was proposed and discussed for ℓ_1-type regularization functions in [52]. For completeness we here sketch the central points of their derivation. An interesting result of this derivation is that the Split Bregman method can also be applied to unconstrained problems, in which case it is equivalent to the well known ADMM. For this reason, this appendix contains an introduction to both the Split Bregman method and the ADMM.

In contrast to the PDHG method described in subsection A.2.1, the Split Bregman searches for a solution of a constrained minimization problem of the form

$$\underset{\mathbf{x}}{\operatorname{argmin}} \sum_{r=1}^{R} \mathcal{R}_i(\mathbf{x}) \text{ subj. to} \frac{1}{2} \|\mathbf{S}\mathbf{x} - \mathbf{y}\|_2^2 < \varepsilon^2 \tag{A.18}$$

where $\mathcal{R} : \mathbb{K}^N \rightarrow \mathbb{R}$ is a regularization function. Solutions of the constrained problem often have advantages over the solutions of the corresponding unconstrained problems. For instance, they involve a less severe trade-off between the data-fidelity term and the regularization term, which often leads to a solution with less bias. Another strength of the Split Bregman method is its ability to handle multiple regularization functions. For this reason, we explicitly consider the case that $\mathcal{R} = \sum_i^R \mathcal{R}_i$, where the $\mathcal{R}_i$ are a set of simpler regularization functions.

Bregman iteration

The central concept underlying the Split Bregman method is the Bregman distance. For a given convex functional E, the latter is defined as

$$D_E^{\mathbf{p}}(\mathbf{x}_1, \mathbf{x}_2) = E(\mathbf{x}_1) - E(\mathbf{x_2}) - <\mathbf{p}, \mathbf{x}_1 - \mathbf{x}_2>, \tag{A.19}$$

with $\mathbf{p} \in \partial E(\mathbf{x}_2)$ belonging to the subgradient of E evaluated at $\mathbf{x}_2$. The Bregman distance is no distance in the traditional sense, e.g. because it is not symmetric. Nevertheless it measures the proximity of its arguments in the sense that $D_E^{\mathbf{p}}(\mathbf{x}_1, \mathbf{x}_1) = 0$ and $D_E^{\mathbf{p}}(\mathbf{x}_1, \mathbf{x}_2) \geq D_E^{\mathbf{p}}(\tilde{\mathbf{x}}, \mathbf{x}_2)$ for any $\tilde{\mathbf{x}}$ on the line segment between $\mathbf{x}_1$ and $\mathbf{x}_2$.

As was proposed by Bregman in [183], the Bregman distances can be used for the solution of constrained optimization problems. To do so, we first consider an unconstrained form of the optimization problem:

$$\underset{\mathbf{x}}{\operatorname{argmin}} \mathcal{R}(\mathbf{x}) + \frac{1}{\lambda}\mathcal{D}(\mathbf{x}). \tag{A.20}$$

Here $\mathcal{D}$ models the constraints, such as fidelity to a set of measured data, and $1/\lambda$ is a parameter used to enforce it in the unconstrained setting. In the context of (A.18), $\mathcal{D}$ can be associated with $\mathcal{D}(\mathbf{x}) =$

$1/2 \left\|\mathbf{S}\mathbf{x} - \mathbf{y}\right\|_2^2$. Instead of directly solving this unconstrained problem, Bregman iteration produces a series of estimates according to the iteration

$$\begin{aligned}\mathbf{x}^{k+1} &= \underset{\mathbf{x}}{\operatorname{argmin}}\, D_{\mathcal{R}}^{\mathbf{p}}(\mathbf{x}, \mathbf{x}^k) + \frac{1}{\lambda}\mathcal{D}(\mathbf{x}) \\ &= \mathcal{R}(\mathbf{x}) - \mathcal{R}(\mathbf{x}^k) - <\mathbf{p}^k, \mathbf{x} - \mathbf{x}^k> + \frac{1}{\lambda}\mathcal{D}(\mathbf{x}).\end{aligned} \tag{A.21}$$

An interesting aspect of this iteration is that it converges towards a solution of the constrained minimization problem, as is discussed further in Refs. [52] and [184].

To be able to perform Bregman iteration, one needs to determine a dual vector $\mathbf{p}^{k+1} \in \partial\mathcal{R}(\mathbf{x}^{k+1})$ for each iteration of the algorithm. To find such a vector, we consider the optimality condition for (A.21)

$$0 \in \partial\left(\mathcal{R}(\mathbf{x}) - \mathcal{R}(\mathbf{x}^k) - <\mathbf{p}^k, \mathbf{x} - \mathbf{x}^k> + \frac{1}{\lambda}\mathcal{D}(\mathbf{x})\right) = \partial\mathcal{R}(\mathbf{x}) - \mathbf{p}^k + \frac{1}{\lambda}\nabla\mathcal{D}(\mathbf{x}),$$

where the subdifferential is to be evaluated at $\mathbf{x} = \mathbf{x}^{k+1}$. If one now inserts the condition for $\mathbf{p}^{k+1}$ into the optimality condition, one obtains the update

$$\mathbf{p}^{k+1} = \mathbf{p}^k - \frac{1}{\lambda}\nabla\mathcal{D}(\mathbf{x}^{\mathbf{k+1}}). \tag{A.22}$$

Having found an update for the dual variable, we can now formulate the Bregman iteration scheme for the solution of (A.18). By inserting the corresponding definition for $\mathcal{D}$ and omitting constant terms of the minimization functional, we obtain the following iterative scheme:

$$\begin{aligned}\mathbf{x}^{k+1} &= \underset{\mathbf{x}}{\operatorname{argmin}}\, \frac{1}{2}\left\|\mathbf{S}\mathbf{x} - \mathbf{y}\right\|_2^2 + \lambda\mathcal{R}(\mathbf{x}) - \lambda<\mathbf{p}^k, \mathbf{x} - \mathbf{x}^k> \\ \mathbf{p}^{k+1} &= \mathbf{p}^k - \frac{1}{\lambda}\mathbf{S}^{\mathsf{H}}\left(\mathbf{S}\mathbf{x}^{k+1} - \mathbf{y}\right).\end{aligned} \tag{A.23}$$

At the first glance, this seems like a complicated iteration. Luckily, a more convenient form can be obtained by iteratively inserting the equation for the $\mathbf{p}$-update into itself, which yields $\mathbf{p}^k = -1/\lambda\mathbf{S}^{\mathsf{H}}\sum_{\kappa=0}^{k}\left(\mathbf{S}\mathbf{x}^{\kappa} - \mathbf{y}\right)$. Inserting this into the expression for the x-update and completing the square yields the iteration

$$\begin{aligned}\mathbf{x}^{k+1} &= \underset{\mathbf{x}}{\operatorname{argmin}}\, \frac{1}{2}\left\|\mathbf{S}\mathbf{x} - \mathbf{y}^k\right\|_2^2 + \lambda\mathcal{R}(\mathbf{x}) \\ \mathbf{y}^{k+1} &= \mathbf{y}^k + \mathbf{y} - \mathbf{S}\mathbf{x}^{k+1}.\end{aligned} \tag{A.24}$$

This iterative scheme only involves the solution of an unconstrained minimization problem. After each iteration, the residual noise of the current estimate is added to the data vector for the next iteration.

Unconstrained Split Bregman method & ADMM

Based on the iteration (A.24), a solution to problem (A.18) can be found. However, it should be noted that the corresponding $\mathbf{x}$-update can be challenging, especially in the presence of multiple regularization functions. For this reason, we first now discuss how Bregman iteration can be used to obtain an efficient algorithm for the solution of unconstrained optimization problems. Notably, the resulting algorithm corresponds exactly to the well known alternating direction of multipliers method [176].

To simplify the problem, we first perform a variable splitting with the goal of decoupling the data fidelity term and the regularization terms. Thus, we introduce auxiliary variables $\{\mathbf{v}_i \in \mathbb{K}^N\}_{i=1,\dots,R}$ and replace the $\mathbf{x}$-update problem by the following equivalent problem

$$\underset{\mathbf{x},\{\mathbf{v}_r\}_i}{\operatorname{argmin}} \frac{1}{2} \left\| \mathbf{S}\mathbf{x} - \mathbf{y}^k \right\|_2^2 + \sum_{r=1}^{R} \lambda \mathcal{R}(\mathbf{v}_r) \text{ subj. to } \mathbf{v}_r = \mathbf{x}. \tag{A.25}$$

As before, this constrained optimization can be tackled using Bregman iteration. If we again apply the scheme in (A.24), this yields the iteration

$$\begin{aligned} \mathbf{x}, \{\mathbf{v}_r\}_r &= \underset{\mathbf{x},\{\mathbf{v}_r\}_r}{\operatorname{argmin}} \frac{1}{2} \left\| \mathbf{S}\mathbf{x} - \mathbf{y}^k \right\|_2^2 + \sum_{r=1}^{R} \lambda_r \mathcal{R}(\mathbf{v}_r) + \sum_{r=1}^{R} \frac{\rho_r}{2} \left\| \mathbf{x} + \mathbf{b}_r - \mathbf{v}_r \right\|_2^2 \\ \mathbf{b}_r^{k+1} &= \mathbf{b}_r^k + \mathbf{x}^{k+1} - \mathbf{v}_r^{k+1}, \end{aligned} \tag{A.26}$$

where the ρ_r are suitable parameters to enforce the equality constraints between the optimization variables [184]. In this scheme the data fidelity term and the regularization terms each depend on a different variable. Nevertheless, the joint optimization over all variables is challenging. To circumvent this issue we perform iterative minimization with respect to each variable separately, which leads to the following simplified scheme

$$\begin{aligned} \mathbf{x}^{l+1} &= \underset{\mathbf{x}}{\operatorname{argmin}} \frac{1}{2} \left\| \mathbf{S}\mathbf{x} - \mathbf{y}^k \right\|_2^2 + \sum_{r=1}^{R} \frac{\rho_r}{2} \left\| \mathbf{x} + \mathbf{b}_r^l - \mathbf{v}_r^l \right\|_2^2 \\ \mathbf{v}_r^{l+1} &= \underset{\mathbf{v}_r}{\operatorname{argmin}} \lambda_r \mathcal{R}(\mathbf{v}_r) + \frac{\rho_r}{2} \left\| \mathbf{x}^{l+1} + \mathbf{b}_r^l - \mathbf{v}_r \right\|_2^2 = \operatorname{prox}_{\frac{\lambda_r}{\rho_r}\mathcal{R}_r}(\mathbf{x}^{l+1} + \mathbf{b}^l) \\ \mathbf{b}_r^{l+1} &= \mathbf{b}_r^l + \mathbf{x}^{l+1} - \mathbf{v}_r^{l+1}. \end{aligned} \tag{A.27}$$

Here the $\mathbf{x}$-update can be computed efficiently since it only contains quadratic terms and can thus be reduced to a linear system of equations by computing the corresponding optimality condition. Moreover, the $\mathbf{v}$-updates correspond to the proximal map for the corresponding regularization function, which often has a simple form. The derived alternating minimization scheme thus features an effective decoupling of the data fidelity and the different regularization terms. This makes it a useful tool when multiple regularization terms are used. It can also be used to deal with more complex regularization functions by allowing for more general constraints of the form $\mathbf{v}_r = \mathbf{A}_r\mathbf{x}$ in problem (A.25). Here we did not follow

Algorithm 6: Split Bregman method

Input: Data $\mathbf{y}$, operator $\mathbf{S}$, regularization functions $\{\mathcal{R}_r\}_{r=1,\dots,R}$, parameters $\{\lambda_r\}_{r=1,\dots,R}$ and $\{\rho_r\}_{r=1,\dots,R}$

Output: estimated solution $\mathbf{x}$

begin

$\mathbf{y}^1 \leftarrow \mathbf{y}$

$\mathbf{x}^1, \mathbf{v}_r^1 \leftarrow \mathbf{S}^{\mathsf{H}}\mathbf{y}$ for $r \in \{1, \dots, R\}$

$\mathbf{b}_\mathbf{r}^\mathbf{1} \leftarrow \mathbf{0}$ for $r \in \{1, \dots, R\}$

for $k = 1, ..., K$ **do**

for $l = 1, ..., L$ **do**

$\mathbf{x}^{l+1} \leftarrow \underset{\mathbf{x}}{\operatorname{argmin}} \frac{1}{2} \left\|\mathbf{S}\mathbf{x} - \mathbf{y}^k\right\|_2^2 + \sum_{r=1}^{R} \frac{\rho_r}{2} \left\|\mathbf{x} + \mathbf{b}_r^l - \mathbf{v}_r^l\right\|_2^2$

$\mathbf{v}^{l+1} \leftarrow \operatorname{prox}_{\frac{\lambda_r}{\rho_r}\mathcal{R}_r}(\mathbf{x}^{l+1} + \mathbf{b}^l)$

$\mathbf{b}_r^{l+1} \leftarrow \mathbf{b}_r^l + \mathbf{x}^{l+1} - \mathbf{v}_r^{l+1}$

end

$\mathbf{y}^{k+1} \leftarrow \mathbf{y}^k + \mathbf{y} - \mathbf{S}\mathbf{x}^{L+1}$

end

return $\mathbf{x}^{J+1}$

end

this approach for the sake of simplicity. Nevertheless, we note that this generalized approach turns out to be very useful for many cases where a simple regularization function (e.g. the ℓ_1-norm) is composed with a non-unitary operator (e.g. finite difference operators).

Constrained Split Bregman method

Having found a way to perform the $\mathbf{x}$-update in the Bregman iteration (A.24), the constrained optimization problem (A.18) can now be solved by inserting the derived alternating minimization scheme (A.27) into (A.24). The resulting iterative scheme is summarized in Algorithm 6.