International Journal on Magnetic Particle Imaging
Vol 7, No 2, Article ID 2112001, 15 Pages

Infinite Science Publishing

Research Article

Optimized sampling patterns for the sparse recovery of system matrices in Magnetic Particle Imaging

Mirco Grosser [a,b,*] · Knopp Tobias [a,b]

[a] Section for Biomedical Imaging, University Medical Center Hamburg-Eppendorf, Hamburg, Germany
[b] Institute for Biomedical Imaging, Hamburg University of Technology, Hamburg, Germany
[*] Corresponding author, email: mi.grosser@uke.de

Received 02 July 2021; Accepted 01 October 2021; Published online 08 December 2021

Abstract

In Magnetic Particle Imaging (MPI), the system matrix plays an important role, as it encodes the relationship between particle concentration and the measured signal. Its acquisition requires a time-consuming calibration scan, which can be a limiting factor in practical applications. Calibration time can be reduced using compressed sensing, which exploits the knowledge that the MPI system matrix has a sparse representation in a suitably chosen domain. This work seeks to further enhance sparse system matrix recovery by optimizing the sampling points to the signal class at hand. For this purpose we introduce an experiment design method based on the Bayesian Fisher information matrix. Our technique uses a previously measured system matrix to tailor the sampling pattern to the signal class at hand. Our tests show that the optimized sampling patterns lead to a more accurate system matrix recovery than popular random sampling approaches. Moreover, our tests demonstrate that the optimized sampling patterns are sufficiently robust to enhance the recovery of system matrices for other types of particles or other experimental conditions.

I. Introduction

Magnetic Particle Imaging (MPI) is a promising technique for imaging the distribution of magnetic nanoparticles (SPIOs) *in vivo* [1, 2]. The method offers both high spatial and high temporal resolution. Furthermore, MPI does not require the use of ionizing radiation, which is in contrast to some popular functional imaging techniques such as positron emission tomography. One of the main target applications of MPI is the imaging of the blood flow for the diagnosis of vascular diseases [3–7]. Others include the tracking of surgical instruments using special magnetic markers [8–10] and the mapping of physiological parameters such as tissue temperature [11].

For non-Cartesian MPI, the most popular image reconstruction approach uses a system matrix (SM) to model the MPI signal generation process. With an SM at hand, an inverse problem can be solved to recover the underlying tracer distribution. To obtain the required SM, one can resort to physical models or perform a calibration scan prior to the actual imaging experiment. While the feasibility of the model-based approach has been demonstrated using various magnetization models [12–14], these approaches are not in wide-spread use due to the numerical effort associated with sufficiently realistic models of the particle-dynamics [14]. Instead the calibration based approach is more commonly used in MPI. The latter is very generic since system matrices can be de-

termined for different kinds of tracers [15, 16] and under different conditions (e.g. temperature and viscosity) [17, 18]. Moreover, scanner specific system imperfections, such as field distortions, are intrinsically taken into account. As a matter of fact, the calibration scan is very time consuming and becomes prohibitively expensive for 3d imaging of larger volumes. For instance the 3d system matrices used in this work each had a measurement time of ∼ 30 hours. Such long calibrations times are particularly problematic for scanners, such as the human-sized low-field brain imager presented in [19]. For these kinds of scanners, coil heating is a serious issue, which effectively prevents continuous measurements over such long times.

To reduce calibration times, various techniques have been proposed based on the paradigm of compressed sensing (CS) [20–25]. These techniques exploit the fact that the MPI system matrix has a sparse representation when applying a suitable transformation such as the discrete cosine transform (DCT), the discrete Fourier Transform (DFT) or the discrete Chebyshev transform (DTT). In combination with pseudo-random measurement points it was found that high quality system matrices can be recovered from measurements with more than 10-fold undersampling [20–22].

A common aspect of past works on CS-based SM recovery is the use of pseudo-random sampling schemes, such as the well-known Poisson disk (PD) pattern. This choice is popular, because CS theory provides performance guarantees for the case of random measurements [26]. While being very generic, such sampling patterns do not necessarily yield the best recovery results for a given measurement matrix and a given signal class to be recovered. For instance, it is well known that, for magnetic resonance imaging (MRI) measurements, variable density patterns yield superior results.

In MRI, aforementioned observation has led to the development of dedicated techniques, which tailor sampling patterns to a given anatomy at hand [27–29]. For instance, OEDIPUS adapts sampling patterns to the sparsity structure of the underlying images [30] by optimizing the constrained Cramer-Rao bound [31]. Similarly, the problem of designing efficient sampling schemes has been addressed for other applications such as network monitoring [32], localization and tracking [33] and graph signal processing [34]. This broad range of applications has also triggered a number of more general works, which provide general frameworks for optimizing the selection or placement of sensors [35–37].

The aim of this work is to investigate the use of optimized sampling patterns for the recovery of MPI system matrices. Our main contribution is the introduction of a new algorithm to optimize sampling patterns based on the sparsity patterns of a previously measured system matrix. The proposed method can be viewed as a Bayesian extension of the OEDIPUS framework. This allows us to incorporate further information about the structure of the system matrices to be recovered. A preliminary version of our framework was described in [38], in the context of MRI image reconstruction. We note however, that this preliminary work does not take into account the intricacies of the MPI calibration process. Our results demonstrate that the optimized patterns yield a more accurate system matrix recovery than the popularly used pseudo-random sampling patterns. In particular, the recovery error is reduced for frequency components with high mixing order, which are often corrupted by a large amount of measurement noise.

II. MPI Theory

II.I. Mathematical Notation

In this work, we use standard mathematical notation where small boldface letters denote vectors and capital boldface letters denote matrices. Moreover, we use the symbol * to denote complex conjugation of either a scalar, a vector or a matrix. In the case of a vector or matrix, complex-conjugation is understood to be performed component-wise. For a matrix $\boldsymbol{A}$, we use $\boldsymbol{A}^\mathsf{T}$ to denote the transposed matrix and $\boldsymbol{A}^\mathsf{H} = (\boldsymbol{A}^*)^\mathsf{T}$ to denote the complex-conjugated and transposed matrix. For a vector $\boldsymbol{x} \in \mathbb{C}^N$, the transposed vector $\boldsymbol{x}^\mathsf{T}$ refers to the corresponding row vector obtained when interpreting $\boldsymbol{x}$ as a $(N \times 1)$-matrix.

II.II. Prerequisites

In MPI the distribution of tracer material $c(\mathbf{r})$ is encoded using a static magnetic gradient field, termed the selection field, which contains a field-free point (FFP) in the origin. The FFP is moved around within the field of view (FOV) by a dynamic, nearly homogeneous field termed the drive field. The region covered by the FFP is commonly refered to as the drive-field FOV (DF-FOV). Note however, that the actual FOV is usually chosen larger than the DF-FOV to avoid artifacts due to signals generated by particles outside the FOV [39].

In order to model the relationship between $c(\mathbf{r})$ and the measured MPI signal, one typically makes the assumption that particle-particle interactions can be neglected. In this case, the Fourier coefficients of the measured voltage signal at frequency f_k can be expressed using a linear model

$$u_k = \int_{\mathrm{FOV}} s_k(\mathbf{r}) c(\mathbf{r}) \mathrm{d}^3\mathrm{r}. \tag{1}$$

Here $\mathbf{r}$ denotes the spatial position and s_k is the k^{th} frequency component of the MPI system function. If we discretize space into N voxels with centers $\{\mathbf{r}_n\}_{n=1,\cdots,N}$ we can formulate the discrete forward model

$$\mathbf{u} = \mathbf{S}\mathbf{c}. \tag{2}$$

10.18416/ijmpi.2021.2112001

Here $\mathbf{u} = (u_k)_{k=1,\cdots,K}$ and $\mathbf{c} = (c(\mathbf{r}_n))_{n=1,\cdots,N}$ denote the measured signal and the particle concentration, respectively. Furthermore, $\mathbf{S} = (w_n \hat{s}_k(\mathbf{r}_n))_{k=1,\cdots,K;n=1,\cdots,N}$ is the MPI system matrix and $(w_n)_{n=1,\cdots,N}$ are quadrature weights induced by the discretization.

The standard procedure for measuring a system matrix is to place a small sample, filled with tracer, in the center of each voxel on the imaging grid and to measure the response of the system. Let c_0 denote the tracer concentration in the sample and $\mathbf{e}_j \in \mathbb{R}^N$ the j^{th} unit vector. Then the described measurement yields $c_0\mathbf{S}\mathbf{e}_j$, which corresponds exactly to the j^{th} column of the system matrix scaled by c_0. In practice, one typically averages multiple measurements in order to obtain a measurement with high signal-to-noise ratio (SNR). As outlined before, this procedure can be quite time-consuming, especially for large 3-dimensional imaging grids. CS-based approaches try to alleviate such problems by measuring the system response only for a subset of all voxels in the imaging grid. The remaining information is then recovered based on prior knowledge, as discussed in Section II.III.

II.III. CS-based System Matrix Recovery

Having performed an undersampled calibration scan, as outlined in the previous section, sparse system matrix recovery aims to recover fully sampled system matrix rows by exploiting the transform domain sparsity of the frequency components. To formalize the underlying ideas, let us consider a fixed row k of the system matrix and let $\mathbf{s} \in \mathbb{C}^N$ denote the vectorized representation of this row. Here and in the following, we omit the row index k for the sake of readability. When extracting the corresponding row from the calibration measurements, one obtains a vector $\mathbf{y} \in \mathbb{C}^M$, which can be related to the underlying system matrix row via a linear measurement model

$$\mathbf{y} = \mathbf{H}_\Omega \mathbf{s} + \mathbf{n}_\Omega. \tag{3}$$

Here $\mathbf{H}_\Omega \in \mathbb{C}^{M\times N}$ denotes the measurement matrix and $\mathbf{n}_\Omega$ denotes the measurement noise, which we assume to be Gaussian with a diagonal covariance matrix $\mathbf{R}_\Omega = \operatorname{diag}(r_1, \ldots, r_M)$.

The measurement matrix can be viewed as a collection of M individual measurements $\{\mathbf{h}_j^\mathsf{T} \in \mathbb{C}^{1\times N}\}_{j\in\Omega}$ each of which constitutes one row of $\mathbf{H}_\Omega$. Here Ω denotes a set of indices each of which is associated with one measurement performed. For later reference, we also introduce the set $\Gamma = \{1, \ldots, N\}$ containing the indices of all possible measurements. For the MPI system matrix, each individual measurement corresponds to the measurement at one voxel-center in the FOV. Thus, the corresponding measurements can be written as $\mathbf{h}_j^\mathsf{T} = \mathbf{e}_j^\mathsf{T}$. A special case is that of full sampling, where $\Omega = \Gamma$ and the corresponding measurement matrix is just the identity matrix.

A further assumption underlying CS is the sparsity of the signal to be recovered. Here, we assume that the signal is S-sparse in a suitably chosen transform domain. To be precise we assume that the signal can be written as $\mathbf{s} = \mathbf{\Psi}^\dagger \boldsymbol{\alpha}$ with $\boldsymbol{\alpha}$ containing only $S < N$ non-zero coefficients. Here $\mathbf{\Psi} \in \mathbb{C}^{Q\times N}$ is a left-invertible sparsifying transform with pseudo-inverse $\mathbf{\Psi}^\dagger$. For the MPI system matrix, common sparsifying transforms are the DCT, the DFT and the DTT.

Following the outlined ideas, CS theory establishes that an estimator for $\mathbf{s}$ can be obtained by solving an ℓ_1-regularized minimization problem of the form

$$\operatorname*{argmin}_{\mathbf{s}} \|\mathbf{\Psi}\mathbf{s}\|_1 \text{ subj. to } \frac{1}{2}\left\|\mathbf{H}_\Omega\mathbf{s} - \mathbf{y}\right\|_2^2 < \varepsilon^2. \tag{4}$$

Here the parameter ε determines the trade-off between the ℓ_1-term and data-fidelity.

CS theory provides recovery guarantees for measurement matrices which fulfill a restricted isometry property [26]. In particular, this property is fulfilled by random sampling patterns. For this reason, pseudo-random patterns, such as Poisson disk patterns are most frequently used. Albeit being very generic, such sampling patterns often do not yield the best results for a given class of signals. This motivates us to investigate optimized sampling patterns in the subsequent sections.

In this work, we solve the CS problem (4) using the Split Bregman method [25, 40]. This method approximates a solution to the constrained problem (4) by solving a series of unconstrained problems according to the iteration

$$\begin{aligned} \mathbf{s}^{i+1} &\leftarrow \operatorname*{argmin}_{\mathbf{s}} \frac{1}{2}\left\|\mathbf{H}_\Omega\mathbf{s} - \mathbf{y}^i\right\|_2^2 + \lambda\|\mathbf{\Psi}\mathbf{s}\|_1 \\ \mathbf{y}^{i+1} &\leftarrow \mathbf{y}^i + \mathbf{y} - \mathbf{H}_\Omega\mathbf{s}^{i+1}. \end{aligned} \tag{5}$$

Here λ is the regularization parameter for the ℓ_1-term in the unconstrained problems, which determines the trade-off between data fidelity and sparsity of the solution. For further details about the method, we refer the reader to Reference [40].

III. Sampling Pattern Optimization

In this section, we introduce our method for the optimization of sampling patterns informed by the sparsity structure of a previously measured system matrix. We start by reviewing a general framework for experiment design based on the Fisher information matrix in Section III.I. Albeit being very generic, this framework itself does not take into account the sparsity of the signals to be recovered. In Section III.II, we outline how the latter can be incorporated by a suitable extension of the framework. In the following Sections III.III and III.IV, we

discuss the solution of the proposed optimization problem and some aspects of the numerical implementation. Finally, Section III.V is concerned with the application of the proposed method to the problem of MPI system calibration.

III.I. Experiment Design

From a statistical point of view, solving the CS problem (4) yields an estimator for $\mathbf{s}$ given measured data $\mathbf{y}$. In the following, let this estimator be denoted by $\hat{\mathbf{s}} : \mathbb{C}^M \to \mathbb{C}^N$. The achievable accuracy of such an estimator strongly depends on the measurement matrix $\mathbf{H}_\Omega$. Experiment design is concerned with the question how to chose the set of measurements $\Omega \subset \Gamma$ such that the estimation error for the underlying parameter vector $\mathbf{s}$ is minimized. In the literature this is also known as the sensor selection problem [36, 37].

In order to assess the achievable accuracy for a given experiment, one can resort to the Cramer-Rao bound (CRB). The latter gives a lower bound for the covariance matrix $\mathrm{cov}(\hat{\mathbf{s}}(\mathbf{y}))$ of an unbiased estimator for $\mathbf{s}$:

$$\mathrm{cov}(\hat{\mathbf{s}}) \succeq \mathbf{J}_\Omega^{-1}(\hat{\mathbf{s}}) = \left(\mathbf{H}_\Omega^{\mathsf{H}} \mathbf{R}_\Omega^{-1} \mathbf{H}_\Omega\right)^{-1}. \tag{6}$$

The matrix $\mathbf{J}_\Omega(\hat{\mathbf{s}}) = \mathbf{H}_\Omega^{\mathsf{H}} \mathbf{R}_\Omega^{-1} \mathbf{H}_\Omega$ is also referred to as the Fisher information matrix (FIM). Moreover, the notation $\succeq$ refers to the Loewner ordering. Thus, the meaning of (6) is that $\mathrm{cov}(\hat{\mathbf{s}}(\mathbf{y})) - \mathbf{J}_\Omega^{-1}(\hat{\mathbf{s}})$ is always a positive semi-definite matrix. Importantly, this implies that the variance of estimates for a parameter s_j can be lower bounded by the corresponding diagonal element $(\mathbf{J}_\Omega^{-1}(\hat{\mathbf{s}}))_{jj}$.

Aforementioned relationships makes it attractive to use $\mathbf{J}_\Omega^{-1}(\hat{\mathbf{s}})$ as a criterion to optimize the set of measurements to be performed. This is captured in the optimization problem

$$\Omega = \underset{\substack{\Omega \subset \Gamma \\ |\Omega| = M}}{\operatorname{argmin}} \; F\left(\mathbf{J}_\Omega^{-1}(\hat{\mathbf{s}})\right). \tag{7}$$

According to this formulation, one searches for a set of M measurement vectors, such that the resulting inverse FIM is small in some sense, which is determined by the function $F : \mathbb{C}^{N \times N} \to \mathbb{R}$. There exist many popular choices for its choice. One is the log-det penalty $F_D(\cdot) = \log\det(\cdot)$, which leads to the notion of D-optimality [41]. This method has the interpretation that the volume of the confidence-ellipsoid for the unknown parameters is minimized. Another popular choice is the function $F_A(\cdot) = \mathrm{tr}(\cdot)$, which leads to the concept of A-optimality [41]. Solving the corresponding optimization problem leads to a design with minimum average variance. In the remainder of this work, we only consider the average variance penalty F_A.

III.II. Oracle-Based Experiment Design

The concepts introduced in the previous section, provide us with a framework to evaluate and to optimize sampling schemes under the linear model. However, this framework does not take into account the transform domain sparsity of the parameters to be estimated. The latter is a key assumption of CS. A second issue is that the CRB is only valid for unbiased estimators. This is problematic because CS reconstruction schemes rely on regularization, which implies that the resulting estimators are usually biased.

To address these issues, we follow the approach used by OEDIPUS to inform experiment design about the sparsity structure of the parameters of interest. The latter is based on the constrained CRB for sparse parameter vectors, which was introduced by Ben-Haim and Eldar in [31]. A key result of this work is that the CRB for an S-sparse parameter vector is equal to the CRB of an estimator bestowed with the oracle knowledge about the location of the non-zero entries in $\boldsymbol{\alpha}$. Therefore, let $\boldsymbol{\alpha}_S$ denote the vector containing only the S non-zero coefficients of $\boldsymbol{\alpha}$. Moreover, let $\mathbf{U} \in \mathbb{R}^{Q \times S}$ be the matrix, which places the coefficients $\boldsymbol{\alpha}_S$ at the corresponding locations in $\boldsymbol{\alpha}$ (i.e. $\boldsymbol{\alpha} = \mathbf{U}\boldsymbol{\alpha}_S$). In this setting, the measured signals can be written as

$$\mathbf{y} = \mathbf{H}_\Omega \boldsymbol{\Psi}^\dagger \mathbf{U} \boldsymbol{\alpha}_S + \mathbf{n}_\Omega, . \tag{8}$$

The matrix $\mathbf{H}_\Omega \boldsymbol{\Psi}^\dagger \mathbf{U} \in \mathbb{C}^{M \times S}$ can thus be viewed as a generalized measurement, which generates the measured data $\mathbf{y}$ from the significant coefficients of the sparse representation of $\mathbf{s}$. Based on this, OEDIPUS optimizes an oracle based CRB of the form

$$\mathrm{cov}(\hat{\boldsymbol{\alpha}}_S) \succeq \mathbf{J}_\Omega^{-1}(\hat{\boldsymbol{\alpha}}_S) = \left(\mathbf{U}^{\mathsf{H}} \left(\boldsymbol{\Psi}^\dagger\right)^{\mathsf{H}} \mathbf{H}_\Omega^{\mathsf{H}} \mathbf{R}_\Omega^{-1} \mathbf{H}_\Omega \boldsymbol{\Psi}^\dagger \mathbf{U}\right)^{-1}. \tag{9}$$

Following our previous line of thought, (9) can be viewed as the CRB for an estimator which only considers the significant coefficients of $\boldsymbol{\alpha}$.

In this work, we extend the OEDIPUS approach by moving to a Bayesian setting similar to that considered in [37]. Thus, we assume that the parameters $\boldsymbol{\alpha}_S$ have a Gaussian prior distribution, $\boldsymbol{\alpha}_S \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Sigma}_{\boldsymbol{\alpha}})$ with covariance matrix $\boldsymbol{\Sigma}_{\boldsymbol{\alpha}}$. In order to optimize sampling patterns, we propose to optimize the inverse Bayesian FIM

$$\tilde{\mathbf{J}}_\Omega^{-1}(\hat{\boldsymbol{\alpha}}_S) = \left(\boldsymbol{\Sigma}_{\boldsymbol{\alpha}}^{-1} + \mathbf{U}^{\mathsf{H}} \left(\boldsymbol{\Psi}^\dagger\right)^{\mathsf{H}} \left(\mathbf{H}_\Omega^{\mathsf{H}} \mathbf{R}_\Omega^{-1} \mathbf{H}_\Omega\right) \boldsymbol{\Psi}^\dagger \mathbf{U}\right)^{-1}. \tag{10}$$

This can be viewed as a Bayesian generalization of the constrained CRB. In fact it can be shown that (10) is the covariance matrix of the minimum mean squared error estimator for the oracle based measurement model (8) and the assumed prior on $\boldsymbol{\alpha}_S$ [42]. Assuming a Gaussian prior distribution might seem like a pretty strict assumption. However, we note that this assumption is commonly made as a first attempt in Bayesian statistical modeling. For example, the commonly used ridge

regression model can be obtained by assuming a linear measurement model with normally distributed measurement noise and a normal prior distribution.

Introducing this Bayesian formulation comes with multiple advantages. First of all, the prior covariance matrix $\mathbf{\Sigma}_{\boldsymbol{\alpha}}$ can be used to incorporate additional knowledge about the signals to be recovered. This is discussed further in Section III.V for our specific problem at hand. Secondly, the introduction of a prior guarantees that $\tilde{\mathbf{J}}_{\Omega}^{-1}(\hat{\boldsymbol{\alpha}}_S)$ is well-defined even for the case that no measurements are performed. Thus, optimization of sampling patterns based on $\tilde{\mathbf{J}}_{\Omega}^{-1}(\hat{\boldsymbol{\alpha}}_S)$ can be performed using both forward and backward-selection approaches. This topic is discussed further in Section III.III.

Before formulating the proposed sampling pattern optimization problem, we note that the approach described in this section requires prior knowledge about the sparsity structure of the signals to be recovered. To obtain such an estimate, the method assumes the availability of a number of K_T representative training signals. In practice, each training signal has a different sparsity structure, which leads to a different form of the matrices $\mathbf{U}$, $\mathbf{R}$ and $\mathbf{\Sigma}_{\boldsymbol{\alpha}}$ and thus to a different form of $\tilde{\mathbf{J}}_{\Omega}^{-1}$. Following this argumentation, our proposed method aims to minimize the average variance of the oracle based estimators. Thus, we are searching for a solution of the problem

$$\Omega = \underset{\substack{\Omega\subset\Gamma \\ |\Omega|=M}}{\operatorname{argmin}} \sum_{k=1}^{K_T} \operatorname{tr}\big(\tilde{\mathbf{J}}_{\Omega,k}^{-1}(\hat{\boldsymbol{\alpha}}_S)\big), \tag{11}$$

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 $\mathbf{U}_k$, $\mathbf{R}_{\Gamma,k}$ and $\mathbf{\Sigma}_{\boldsymbol{\alpha},k}$, which refer to the respective $\mathbf{U}$, $\mathbf{R}_\Gamma$ and $\mathbf{\Sigma}_{\boldsymbol{\alpha}}$ matrices associated with the k^{th} training signal.

III.III. Solution of the Sensor Selection Problem

Solving discrete experiment design problems like (11) is a difficult task because the problem at hand is a non-convex integer programming problem. For their solution, different methods have been proposed. Some of them reformulate the problem, such that it can be solved using semi-definite programming [35–37]. Others employ greedy algorithms to iteratively add and or subtract measurements from a given set of measurements [43, 44]. OEDIPUS uses a backwards selection algorithm, which starts with $\Omega = \Gamma$ and iteratively removes measurements until Ω contains the target number of measurements [30]. In this work, we use a forward selection algorithm, which starts with an empty set and iteratively adds measurements until the target number of measurements is reached [37]. For undersampling factors larger than 2 this is advantageous because the number of measurements to add to Ω becomes smaller than the number of samples one would need to remove from the set of all candidate measurements Γ.

Algorithm 1: Selecting one measurement

Input: set of candidate measurements $\tilde{\Gamma}$
current set of measurements Ω
inverse FIMs $\{\tilde{\mathbf{J}}_{\Omega,k}^{-1}\}_{k=1}^{K_T}$
matrices $\mathbf{H}_\Gamma$, $\mathbf{\Psi}$ and $\{\mathbf{U}_k\}_{k=1}^{K_T}$
covariance matrices $\{\mathbf{R}_{\Gamma,k}\}_{k=1}^{K_T}$
and $\{\mathbf{\Sigma}_{\boldsymbol{\alpha},k}\}_{k=1}^{K_T}$

1: find the sensor $j \in \tilde{\Gamma}$, which minimizes $\sum_{k=1}^{K_T} \operatorname{tr}(\tilde{\mathbf{J}}_{\Omega,k}^{-1}(\hat{\boldsymbol{\alpha}}_S))$
2: add j to Ω and update $\{\tilde{\mathbf{J}}_{\Omega,k}^{-1}(\hat{\boldsymbol{\alpha}}_S)\}_{k=1}^{K_T}$
3: remove j from $\tilde{\Gamma}$

Output: set of measurements Ω
inverse FIMs $\{\tilde{\mathbf{J}}_{\Omega,k}^{-1}\}_{k=1}^{K_T}$
remaining candidate measurements $\tilde{\Gamma}$

The procedure for adding one measurement to a given set of measurements Ω is summarized in Algorithm 1. Here the main computational complexity arises from the fact that the change in cost function needs to be computed for each candidate measurement contained in $\tilde{\Gamma}$. When embedding Algorithm 1 into a full optimization algorithm, one typically performs an exhaustive search over all remaining candidates, whenever a measurement is to be added to Ω. Thus, the change in cost function needs to be computed on the order of $\mathcal{O}(MN)$ times. For growing problem sizes, this leads to a significant increase in computation time.

To alleviate this 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 candidate set. Thus, given a temporary measurement set Ω and the corresponding matrices $\{\tilde{\mathbf{J}}_{\Omega,k}^{-1}\}_{k=1}^{K_T}$, we propose the following batch-based procedure for adding more measurements.

1. Compute the change in cost function for all candidates in the candidate set Γ.
2. Form a batch $\tilde{\Gamma}$ containing the B candidates with the largest decrease of the cost function in (11).
3. Iteratively add a number of M_B measurements from the batch $\tilde{\Gamma}$ as described in Algorithm 1.

In order to reach the target number of measurements, the procedure needs to be repeated iteratively. The algorithm obtained in this way is summarized in Algorithm 2. Here the change in cost function needs to be computed on the order of $\mathcal{O}(\frac{MN}{M_B} + MB)$ times, which can be a quite significant reduction depending on the chosen values of B and M_B.

Algorithm 2: 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 $\{\mathbf{R}_{\Gamma,k}\}_{k=1}^{K_T}$
and $\{\mathbf{\Sigma}_{\boldsymbol{\alpha},k}\}_{k=1}^{K_T}$
number of measurements M
batch parameters B and M_B

$\Omega \leftarrow \{\}$
$\{\tilde{\mathbf{J}}_{\Omega,k}^{-1}\}_{k=1}^{K_T} \leftarrow \{\mathbf{\Sigma}_{\boldsymbol{\alpha},k}\}_{k=1}^{K_T}$
for $i = 1,\dots,\frac{M}{M_B}$ **do**
  create a batch $\tilde{\Gamma}$ containing the B candidates with the largest decrease of $\sum_{k=1}^{K_T} \operatorname{tr}(\tilde{\mathbf{J}}_{\Omega,k}^{-1}(\hat{\boldsymbol{\alpha}}_S))$
  for $k = 1,\dots,M_B$ **do**
    add one candidate from $\tilde{\Gamma}$ to Ω using Algorithm 1
  end for
end for

Output: set of measurements Ω

The choice of the batch size B and the number of measurements to be added per batch M_B are important parameters to be set, when using the proposed method. As a relevant limit, we note that the regular sequential forward selection algorithm, is obtained when choosing $M_B = 1$ or $B = |\Omega|$. On the other hand, one would ideally like to pick M_B as large as possible and B as small as possible in order to minimize computation times. A good strategy for choosing B and M_B is to perform a number of small test runs, where only a small number of measurements are selected. By comparing the results with the case $M_B = 1$, one can determine a suitable set of parameters, such that the computation time is minimized without compromising the quality of the obtained measurement set Ω.

III.IV. Numerical Implementation

As outlined before, the main numerical challenge in Algorithm 1 is computing the change in cost function $\operatorname{tr}(\tilde{\mathbf{J}}_\Omega^{-1}(\hat{\boldsymbol{\alpha}}_S))$ for all candidate measurements. Moreover, the inverse FIM $\tilde{\mathbf{J}}_\Omega^{-1}(\hat{\boldsymbol{\alpha}}_S)$ needs to be updated, once the best candidate measurement has been identified. A naive computation of these quantities would require the inversion of multiple $S \times S$ matrices. This leads to a high computational complexity of $\mathcal{O}(K_T|\tilde{\Gamma}|S^3)$, which renders the algorithm infeasible for large problem sizes.

To perform aforementioned computations efficiently, we exploit the diagonal form of the noise-covariance matrix. Under this assumption, (10) can be written as [30, 37]

$$\tilde{\mathbf{J}}_\Omega^{-1}(\hat{\boldsymbol{\alpha}}_S) = \Big(\mathbf{\Sigma}_{\boldsymbol{\alpha}}^{-1} + \sum_{j\in\Omega} \frac{\mathbf{h}_{\boldsymbol{\alpha},j}^* \mathbf{h}_{\boldsymbol{\alpha},j}^\mathsf{T}}{r_j}\Big)^{-1}. \tag{12}$$

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 [30, 37, 43] and apply the Sherman-Morrison-Woodbury matrix inversion lemma [45] to (12). 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

$$\tilde{\mathbf{J}}_{\Omega+j}^{-1}(\hat{\boldsymbol{\alpha}}_S) = \tilde{\mathbf{J}}_\Omega^{-1}(\hat{\boldsymbol{\alpha}}_S) - \frac{\tilde{\mathbf{J}}_\Omega^{-1}(\hat{\boldsymbol{\alpha}}_S)\mathbf{h}_{\boldsymbol{\alpha},j}^* \mathbf{h}_{\boldsymbol{\alpha},j}^\mathsf{T} \tilde{\mathbf{J}}_\Omega^{-1}(\hat{\boldsymbol{\alpha}}_S)}{r_j + \mathbf{h}_{\boldsymbol{\alpha},j}^\mathsf{T} \tilde{\mathbf{J}}_\Omega^{-1}(\hat{\boldsymbol{\alpha}}_S)\mathbf{h}_{\boldsymbol{\alpha},j}^*}. \tag{13}$$

Hence, matrix inversion can be circumvented by computing the matrix-vector products on the right hand side of (13).

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

$$\operatorname{tr}(\tilde{\mathbf{J}}_{\Omega+j}^{-1}(\hat{\boldsymbol{\alpha}}_S)) = \operatorname{tr}(\tilde{\mathbf{J}}_\Omega^{-1}(\hat{\boldsymbol{\alpha}}_S)) - \frac{\mathbf{h}_{\boldsymbol{\alpha},j}^\mathsf{T} \tilde{\mathbf{J}}_\Omega^{-2}(\hat{\boldsymbol{\alpha}}_S)\mathbf{h}_{\boldsymbol{\alpha},j}^*}{r_j + \mathbf{h}_{\boldsymbol{\alpha},j}^\mathsf{T} \tilde{\mathbf{J}}_\Omega^{-1}(\hat{\boldsymbol{\alpha}}_S)\mathbf{h}_{\boldsymbol{\alpha},j}^*}. \tag{14}$$

Thus, the computational complexity of Algorithm 1 is reduced to $\mathcal{O}(K_T|\tilde{\Gamma}|S^2)$.

III.V. Application to MPI System Matrix Calibration

To generate optimized sampling patterns for MPI system matrices, we propose to use an exisiting fully sampled system matrix as training data. The adapted pattern can then be used for the accelerated 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 instance when using new particles, performing measurements under different physical conditions (e.g. different temperature) or when changing acquisition parameters such as the drive-field amplitude.

Due to the MPI measurement process, all frequency components of the system matrix are acquired jointly using the same sampling pattern. Thus, the training signals should encompass 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 that the selection of frequency components approximately covers the range of sparsity patterns that occur. If this is not the case, the optimized patterns might not perform well on unseen data. 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

Algorithm 3: 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

$\Gamma \leftarrow \{1,2,\dots,N\}$
$\tilde{\mathbf{S}} \leftarrow$ representative subset of K_T rows from $\mathbf{S}$
$\{\mathbf{U}_k\}_{k=1}^{K_T} \leftarrow$ sparsity patterns of the rows in $\tilde{\mathbf{S}}$
$\mathbf{\Sigma_s} \leftarrow$ signal variances for each voxel
$\{\mathbf{\Sigma}_{\boldsymbol{\alpha},k}\}_{k=1}^{K_T} \leftarrow \mathbf{U}_k^{\mathsf{H}}\left(\mathbf{\Psi}^\dagger\right)^{\mathsf{H}}\mathbf{\Sigma_s}\mathbf{\Psi}^\dagger\mathbf{U}_k$
$\{\mathbf{R}_{\Gamma,k}\}_{k=1}^{K_T} \leftarrow$ noise variances of $\tilde{\mathbf{S}}$
$\Omega \leftarrow$ choose M measurements using Algorithm 2

Output: set of measurements Ω

with an SNR threshold. The latter ensures that the chosen frequency components contain a sufficient amount of information.

A mechanism to incorporate further knowledge about the signal class at hand is the prior covariance matrix $\mathbf{\Sigma}_{\boldsymbol{\alpha}}$. In this work, we chose $\mathbf{\Sigma}_{\boldsymbol{\alpha}}$ based on the observation that, for MPI system matrices, the signal intensities are typically largest in the center of the DF-FOV. For positions outside the DF-FOV, the signal intensities decrease because the field free point does not cover these regions. To model this behavior in image space, we use a diagonal covariance matrix $\mathbf{\Sigma_s}$, with spatially dependent variance. 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

$$\mathbf{\Sigma_s} = \mu \operatorname{diag}\left(\operatorname{var}\left(\{\mathbf{s}_k\}_{k=1}^{K_T}\right)\right), \tag{15}$$

where the variance is calculated componentwise and over the set of K_T frequency components included in the training data. In order to apply it with the proposed Bayesian FIM, $\mathbf{\Sigma_s}$ needs to be transformed into the space spanned by the sparse coefficients $\boldsymbol{\alpha}_S$. Thus, our prior covariance matrix takes the form

$$\mathbf{\Sigma}_{\boldsymbol{\alpha}} = \mathbf{U}^{\mathsf{H}}\left(\mathbf{\Psi}^\dagger\right)^{\mathsf{H}}\mathbf{\Sigma_s}\mathbf{\Psi}^\dagger\mathbf{U}. \tag{16}$$

The full method, obtained when putting everything together, is summarized in Algorithm 3.

IV. Materials and Methods

IV.I. Experiments

In order to evaluate the proposed method, we used both 2d and 3d datasets. In both cases, two system matrices were measured with different tracer materials. Respectively, one of the system matrices was used to generate optimized sampling patterns according to the proposed method. For testing purposes, the other system matrix was undersampled according to the obtained patterns and system matrix recovery was performed using the undersampled data.

All of the system matrices were measured with a Bruker Preclinical MPI scanner. The two 2d system matrices were measured using the MPI tracer perimag (micromod Partikeltechnologie GmbH, Rostock, Germany). The first one was measured using liquid particles, whereas the second one was measured using perimag particles that were immobilized using dental cement. Both system matrices were acquired with a FOV of $(30 \times 30 \times 1)\,\mathrm{mm}^3$ and a grid size of $30 \times 30 \times 1$. The drive-field amplitudes were 12, 12, and 0 mT and the selection field gradients were set to -1, -1, and $2\,\mathrm{T\,m}^{-1}$. In these 2d experiments, the system matrix measured with liquid perimag was used for the generation of sampling patterns. Testing was performed using the system matrix measured with immobilized particles.

Additionally, we used two 3d system matrices, which are contained in the Open MPI Data repository (calibration datasets 6 and 7) [46]. For the first one the tracer material was perimag, whereas synomag-D (micromod Partikeltechnologie GmbH, Rostock, Germany) was used for the second one. Both system matrices were acquired with a FOV of $(37 \times 37 \times 18.5)\,\mathrm{mm}^3$ and a grid size of $37 \times 37 \times 37$. The drive-field amplitudes were 12, 12, and 12 mT and the selection field gradients were -1, -1, and $2\,\mathrm{T\,m}^{-1}$. In the 3d experiments, 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.

IV.II. Sampling Patterns

In our study, the 2d SM with liquid perimag and the 3d SM with synomag particles were employed to generate optimized sampling patterns using the proposed algorithm. For the training we applied an SNR threshold of 3 and made a selection of 30 random frequency components for the 2d case and of 15 random frequency components for the 3d case. As sparsifying transform we chose the DCT-II. To generate sparse approximations, a threshold was applied 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 the prior distribution, we normalized all frequency components by their ℓ_2-norm. The variances for the prior distributions were then calculated according to (16). The proportionality factor was set to $\mu = 3$ for both the 2d and the 3d case.

To fix the noise variances, we assumed a constant

value $r(\mathbf{s}_k)$ 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 $3\times3\times3$ voxels for the 3d case. Now let $\mathbf{s}_k^{\text{bg}}$ contain the values of the given frequency component $\mathbf{s}_k$ in these corner voxels. Then our estimate can be written as

$$r(\mathbf{s}_k) = \frac{\left\| \mathbf{s}_k^{\text{bg}} - \bar{\mathbf{s}}_k^{\text{bg}} \right\|_2^2}{\left\| \mathbf{s}_k^{\text{bg}} \right\|_0 - 1}. \tag{17}$$

Here each entry in $\bar{\mathbf{s}}_k^{\text{bg}}$ corresponds to the mean of the values in the corresponding corners of the FOV.

With the training data at hand, we used Algorithm 3, 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 the use of batching (i.e setting $B = N$ and $M_B = 1$). For the 3d case, we used a batch size of $B = 5000$ and $M_B = 8$.

IV.III. System Matrix Recovery

To test the generated patterns we generated correspondingly undersampled data from the 2d SM with immobilized perimag and the 3d SM with liquid perimag. For the 2d experiment, we generated datasets with undersampling factors of 2, 4, 6, 8 and 10. For the 3d case we consider an undersampling factor of 20. For comparison we also generated 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 (4) for all frequency components with 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 regulariziation parameter was manually chosen such that the mean normalized root mean squared deviation (NRMSD) (see (19)) was minimized. This resulted in a value of $\lambda = 0.005$ for the PD pattern and a value of $\lambda = 0.002$ for the the optimized sampling pattern. In all cases the solver was run for 50 inner and 10 outer iterations.

IV.IV. Experimental Evaluation

To evaluate the quality of the recovered system matrices, we use visual inspection as well as several quantitative error measures. Moreover, we use the recovered SMs for image reconstruction. The resulting images serve as an indicator for the kind of image quality that can be obtained when the recovered system matrices are used in actual applications.

As error measures for the recovered frequency components and for the reconstructed images we use the NRMSD and the structural similarity index (SSIM). For a reference signal $\mathbf{x}$ and an approximation $\mathbf{x}^{\text{approx}}$, the former can be defined as

$$\text{NRMSD}(\mathbf{x}, \mathbf{x}^{\text{approx}}) = \frac{\|\mathbf{x} - \mathbf{x}^{\text{approx}}\|_2}{\sqrt{N}\,\|\mathbf{x}\|_\infty}, \tag{18}$$

with N being the number of elements in $\mathbf{x}$. In contrast to the NRMSD, the SSIM aims to quantify the perceived similarity between both signals [47]. An aspect to keep in mind here is that the SSIM was originally developed for real-valued, non-negative images. Since the MPI system matrix is complex-valued 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 evaluating complete system matrices we use the mean NRMSD of the recovered frequency components as an aggregate error measure

$$\overline{\text{NRMSD}}(\mathbf{S}, \mathbf{S}^{\text{approx}}) = \frac{1}{K}\sum_{k=1}^{K} \text{NRMSD}(\mathbf{s}_k, \mathbf{s}_k^{\text{approx}}). \tag{19}$$

Here k runs over all frequency components. In the same manner, we compute the mean SSIM and denote it by $\overline{\text{SSIM}}$.

For the image reconstruction, we used the 3d datasets of the resolution phantom and the shape phantom contained in the Open MPI Data repository. The drive field amplitudes and selection field gradients for these measurements correspond to the values used for the system matrix measurement. Image reconstruction was performed using the regularized Kaczmarz algorithm with 10 iterations and a relative regularization parameter [48] of 0.0025.

IV.V. Code and Data Availability

Code and example data for the proposed method are available at https://github.com/IBIResearch/OptimizedSystemMatrixSampling.

V. Results

V.I. 2d System Matrices

The covariance matrix $\mathbf{\Sigma}_\mathbf{s}$ is an important parameter of the proposed method as it has a significant influence on

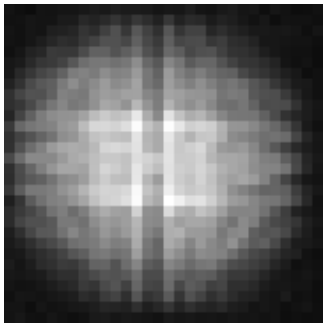

Figure 1: Diagonal of the spatial prior covariance matrix $\mathbf{\Sigma_s}$ obtained from the 30 frequency components used for generating the 2d sampling patterns.

the sampling density of the resulting patterns. In Figure 1, we show its spatial representation $\mathbf{\Sigma_s}$ obtained using (15). 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 2. Especially for the case 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. Besides that, no additional structures or clustering can be observed. Probably, 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 without batching, look almost identical to the ones obtained with batching. This indicates, that the batching approach can indeed be used to accelerate the solution of problem (11) in our given setting, without significantly altering 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, this behavior can clearly be seen. As an illustration, we show a selection of the recovered patterns in Figure 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 $k = 62$.

Since the signal intensity tends to be largest in the DF-FOV, the non-uniform sampling leads to an overall

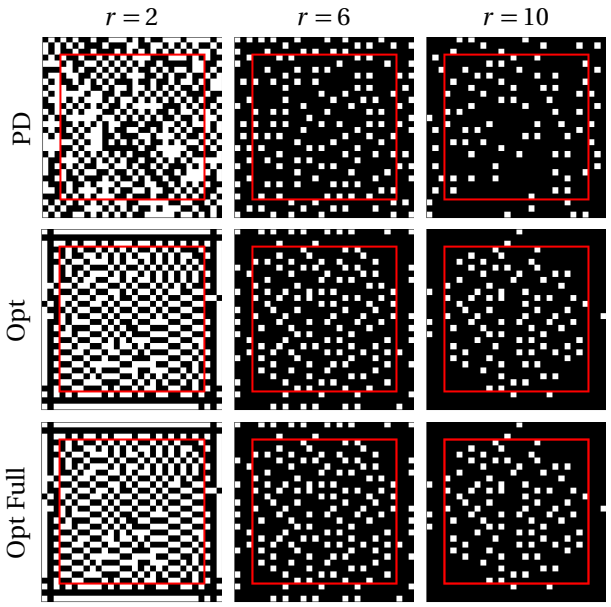

Figure 2: Sampling patterns generated using Poisson disk sampling (top row), the proposed method (middle row) and the proposed method without batching (bottom row). The patterns are shown for undersampling factors 2,6 and 10. For reference, the borders of the DF-FOV are indicated by red lines.

decrease in recovery error. For the plotted frequency components, this is reflected in the NRMSD values superimposed on to 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 4. This plot shows an overall reduction in NRMSD irrespective of the SNR of the frequency components. Interestingly, this reduction in $\overline{\text{NRMSD}}$ can be observed for all regularization parameters considered, as can be seen in the upper panel of Figure 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 sample with high intensity (i.e. with high information content) increases the fewer the number of measurements. This relationship is illustrated in the bottom panel of Figure 5 where we plot the $\overline{\text{NRMSD}}$ in dependence of the undersampling factor.

V.II. 3d System Matrices

For the 3d experiment, the non-uniform distribution of samples becomes even more apparent. As can be seen in Figure 6, the sampling density in the center of the FOV is increased whereas there are 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 7. The residual structures in the error maps for the random

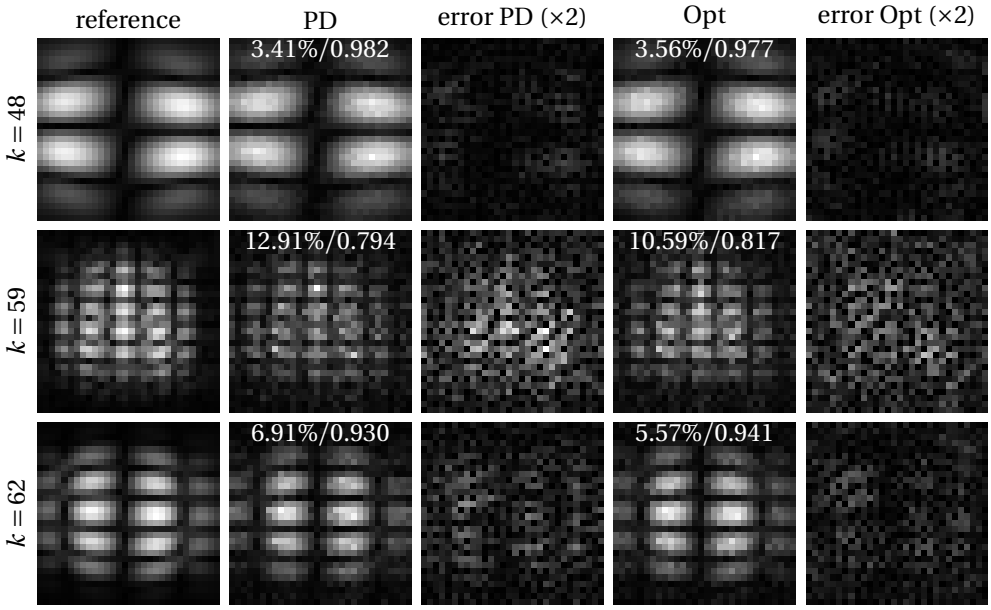

Figure 3: Recovered frequency components from 6-fold undersampled data. The second and third column show the patterns obtained from PD sampling and the respective error plots. The fourth and fifth column show the corresponding quantities for patterns obtained using Algorithm 3. The NRMSD (in %) and the SSIM with respect to the fully sampled reference are superimposed onto the plots. All plots in a given row are scaled to the same colorbar.

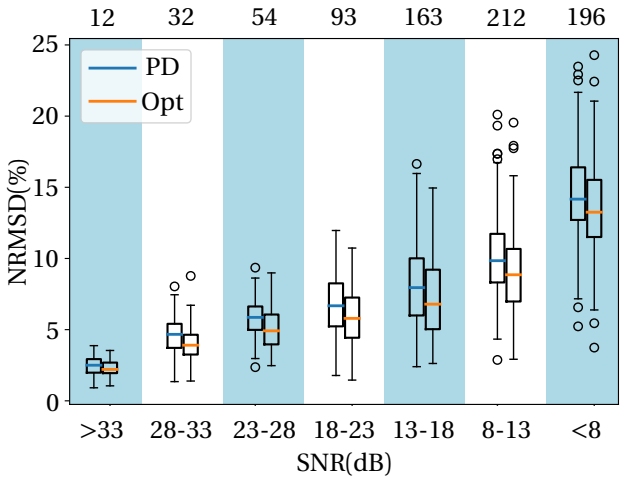

Figure 4: Box-Whiskers plot showing the NRMSD of the 2d-SMs recovered from 6-fold undersampled data. The boxes accumulate the 25% to 75% quantile. The numbers on top of the SNR bins denote the number of frequency components contained in them.

sampling indicate a bias which is absent in the recoveries using the optimized sampling patterns. On the other hand, the left column shows a case where the recovery error in the corners of the FOV is increased compared to the PD sampling pattern.

As can be seen in Figure 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 pattern leads 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 dB. Taking 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 (random sampling) and 0.849 (proposed method).

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 SM contain an increased amount of noise, as can be seen in Figures 9 and 10. For the optimized patterns, the noise is clearly reduced. This improvement is also reflected in the NRMSD and SSIM values on the right side of the figures.

VI. Discussion

Our results show that CS based system matrix recovery can be enhanced by using sampling patterns optimized based on the Bayesian FIM. In contrast to pseudo-random sampling, the optimized sampling patterns display a non-uniform sampling density, which takes into account the increased signal intensity typically observed in the FOV center as opposed to the borders of the FOV. Thus, to the best of our knowledge, this is the first work

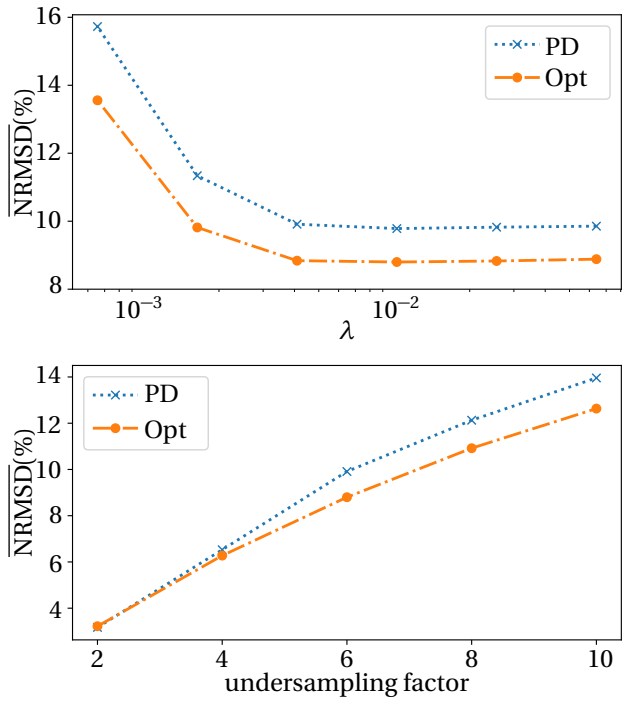

Figure 5: $\overline{\text{NRMSD}}$ of the 2d-SMs in dependence of the regularization parameter λ for 6-fold undersampling (top) and $\overline{\text{NRMSD}}$ in dependence of the undersampling factor (bottom). For each undersampling factor, λ was chosen such that the $\overline{\text{NRMSD}}$ was minimized for the respective method.

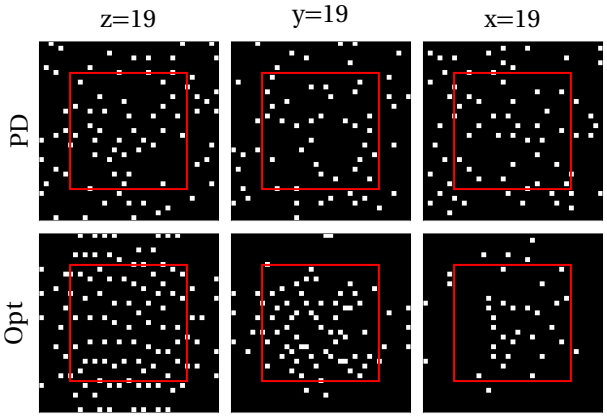

Figure 6: Sampling patterns generated using Poisson disk sampling (top row) and the proposed method (bottom row) for an undersampling factor of 20. The columns show cuts in the xy-, xz-, and yz-plane respectively. For reference, the borders of the DF-FOV are indicated by red lines.

investigating variable density sampling for the recovery of MPI system matrices.

In all of our tests, the optimized sampling patterns resulted in a improved recovery of the system matrices used for testing in terms of the $\overline{\text{NRMSD}}$. A more detailed look reveals that that the optimized pattern can lead to a slightly worse recovery of frequency components with high SNR, whereas the recovery is clearly improved for patterns with a low SNR. An explanation is that frequency components with high SNR are often associated with a low mixing order 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 a more uniform sampling pattern. 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 system matrices 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 being used in a setting that targets high image resolution.

In the proposed method, the degree of non-uniformity can be controlled with the parameter μ, which determines the width of the prior distribution used in our model. 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 of the SM recovery, 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. Since the sampling density is reduced outside the DF-FOV, it follows that increasing the FOV would require only a comparably small number of additional measurements. Such an overscan of the DF-FOV is attractive because it reduces signal leakage from particles outside the DF-FOV [39]. In a multi-patch setting, the overscan can be used to generate overlap between patches. In turn, this leads to an improved image reconstruction when using a suitable multi-patch reconstruction scheme [49–51].

The sampling patterns obtained with the proposed method depend on the sparsity structure of the system matrix used to setup problem (11). One can expect that the generated patterns 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 SMs.

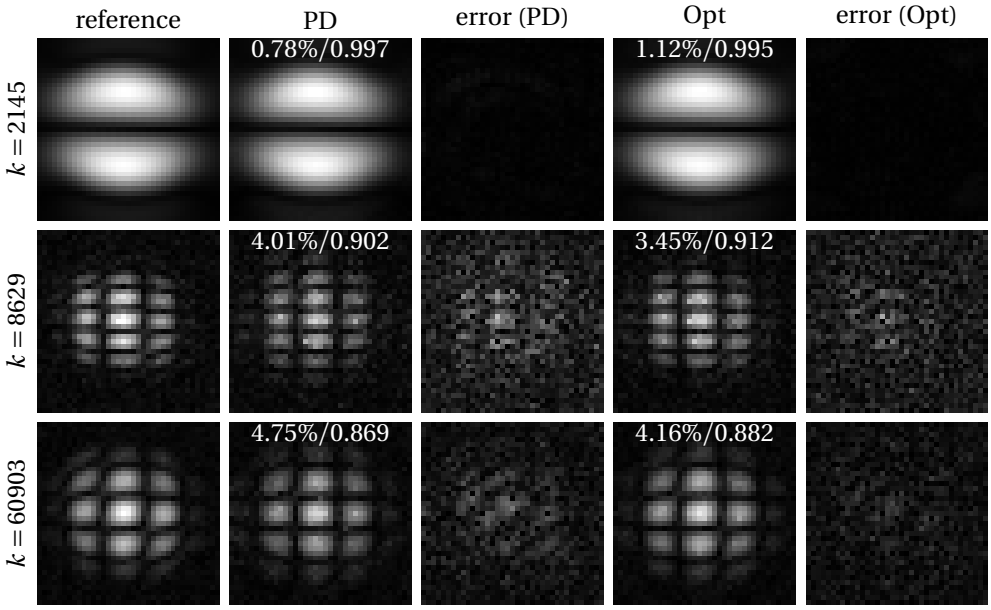

Figure 7: Recovered frequency components from 20-fold undersampled data of a 3d system matrix. The second and third column show the patterns obtained from Poisson disk sampling and the respective error plots. The fourth and fifth column show the corresponding quantities for patterns obtained using Algorithm 3. The NRMSD (in %) and the SSIM with respect to the fully sampled reference are superimposed onto the plots. All plots in a given row are scaled to the same colorbar.

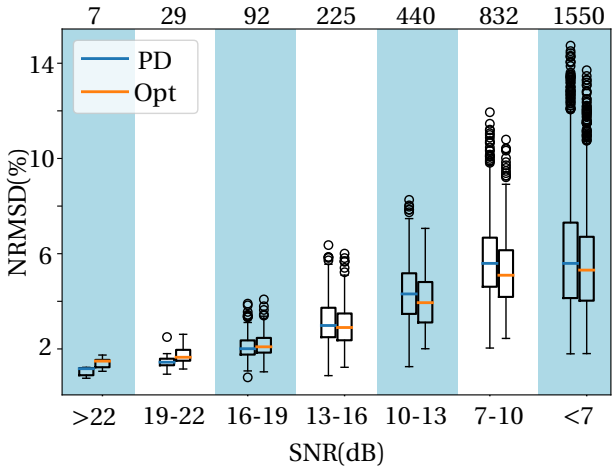

Figure 8: Box-Whiskers plot showing the NRMSD of the recovered 3d-SMs. The boxes accumulate the 25% to 75% quantile. The numbers on top of the SNR bins denote the number of frequency components contained in corresponding bin.

Here an important aspect is that a significant part of the structure of MPI system matrices is determined by the field configuration and the measurement sequence used. For instance, the oscillatory patterns which can be observed in the system matrices used in this work are a general property of FFP-based scanners with a Lissajous-type FFP trajectory [52, 53]. In the DCT domain, these patterns lead to the observation that the non-zero coefficients are contained in a bounded region containing the spatial frequency of the oscillatory pattern. Of course finer details of the sparsity structure are dependent on the particles at hand and in particular on their magnetization curve. A further point to watch when using the proposed method is that the spatial dependence of the signal intensities, which is used in (15), should show a similar behavior for both the system matrix used for training and the one to be recovered. Again, this spatial dependence is strongly dependent on the particles' magnetization curve.

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 [15]. 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

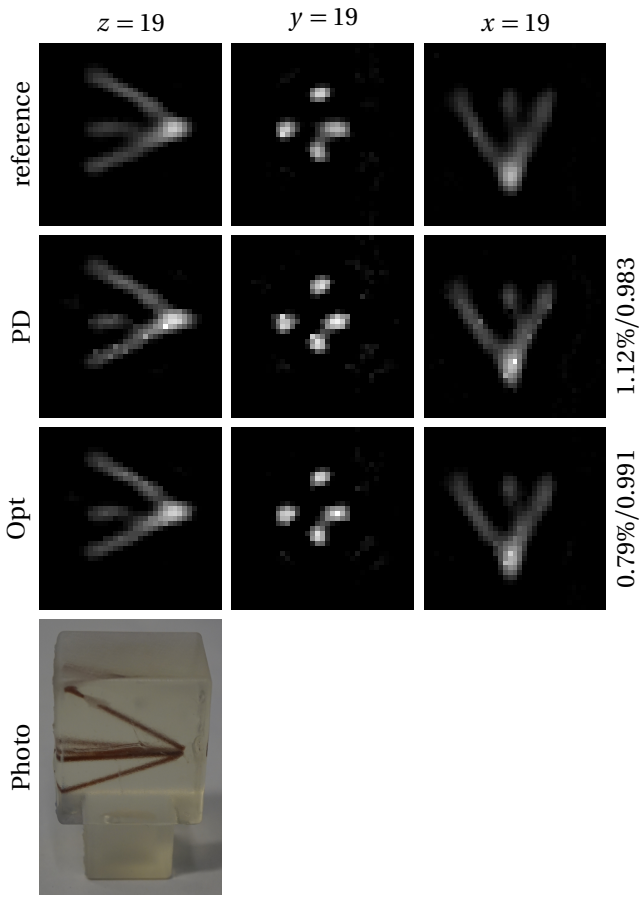

Figure 9: Reconstruction of the Open MPI Data resolution phantom using the recovered 3d system matrices. The columns show cuts in the xy-, xz-, and yz-plane respectively. The numbers next to the reconstructions denote their NRMSD (in %) and their SSIM with respect to the reference reconstruction. All plots in a given column are scaled to the same colorbar.

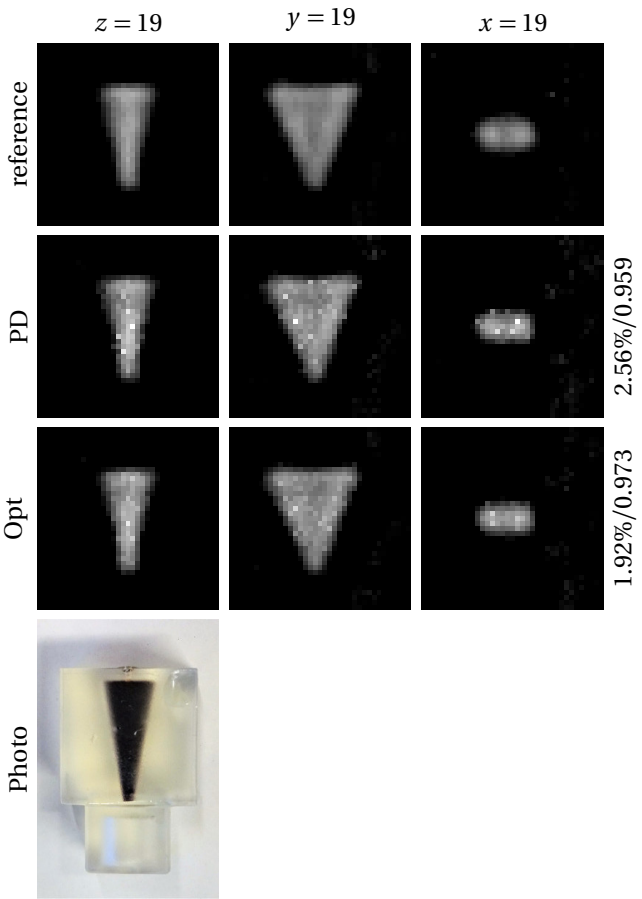

Figure 10: Reconstruction of the Open MPI Data shape phantom using the recovered 3d system matrices. The columns show cuts in the xy-, xz-, and yz-plane respectively. The numbers next to the reconstructions denote their NRMSD (in %) and their SSIM with respect to the reference reconstruction. All plots in a given column are scaled to the same colorbar.

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 [54, 55]. 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.

We envision the proposed method to be used for an offline computation of samplings patterns, which can then be used for the measurement of other system matrices. Therefore, numerical performance is not of prime importance. We note however that the scaling of numerical effort with growing problem sizes is a known issue of many CRB-based sensor-selection approaches. Using the proposed optimization scheme, the computation of sampling patterns took ∼10 s for the 2d SM and ∼7.5 h for the 3d case. The proposed batching scheme lead to a ∼2-fold reduction of computation time for the 2d case and a ∼4.5-fold reduction for the 3d case. All computations were performed using one thread on a workstation equipped with two Intel Xeon CPU E5-2640 v3 CPUs running at 2.6 GHz and a main memory of 512 GB.

The method described in this work can contribute to the acceleration of the calibration process in MPI and thus save valuable time. The approach is very general and we expect that it will prove beneficial for other scanner architectures, such as single-sided MPI scanners, which feature a fairly inhomogeneous sensitivity profile. Another very interesting application is the measurement of system matrices using non-delta samples as proposed in [22]. Such approaches could probably be accelerated

even more using our experiment design framework. Finally, it would be interesting to extend our framework to the joint recovery of system matrices in a multi-patch setting. In such a setting, even larger undersampling factors could be achieved by exploiting the similarity between patches [56]. An extension of our method could help answer the question how to optimally sample the system matrices in such a setting.

References

[1] B. Gleich and J. Weizenecker. Tomographic imaging using the nonlinear response of magnetic particles. *Nature*, 435(7046):1214–1217, 2005, doi:10.1038/nature03808.

[2] J. Weizenecker, B. Gleich, J. Rahmer, H. Dahnke, and J. Borgert. Three-dimensional real-time in vivo magnetic particle imaging. *Physics in Medicine and Biology*, 54(5):L1–L10, 2009, doi:10.1088/0031-9155/54/5/L01.

[3] J. Salamon, M. Hofmann, C. Jung, M. G. Kaul, F. Werner, K. Them, R. Reimer, P. Nielsen, A. vom Scheidt, G. Adam, T. Knopp, and H. Ittrich. Magnetic Particle / Magnetic Resonance Imaging: In-Vitro MPI-Guided Real Time Catheter Tracking and 4D Angioplasty Using a Road Map and Blood Pool Tracer Approach. *PLOS ONE*, 11(6):e0156899M. Yamamoto, Ed., 2016, doi:10.1371/journal.pone.0156899.

[4] P. Ludewig, N. Gdaniec, J. Sedlacik, N. D. Forkert, P. Szwargulski, M. Graeser, G. Adam, M. G. Kaul, K. M. Krishnan, R. M. Ferguson, A. P. Khandhar, P. Walczak, J. Fiehler, G. Thomalla, C. Gerloff, T. Knopp, and T. Magnus. Magnetic Particle Imaging for Real-Time Perfusion Imaging in Acute Stroke. *ACS Nano*, 11(10):10480–10488, 2017, doi:10.1021/acsnano.7b05784.

[5] M. G. 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. *Physics in Medicine & Biology*, 63(6):064001, 2018, doi:10.1088/1361-6560/aab136.

[6] P. Szwargulski, M. Wilmes, E. Javidi, F. Thieben, M. Graeser, M. Koch, C. Gruettner, G. Adam, C. Gerloff, T. Magnus, T. Knopp, and P. Ludewig. Monitoring Intracranial Cerebral Hemorrhage Using Multicontrast Real-Time Magnetic Particle Imaging. *ACS Nano*, 14(10):13913–13923, 2020, doi:10.1021/acsnano.0c06326.

[7] 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. *Scientific Reports*, 11(1):4529, 2021, doi:10.1038/s41598-021-83821-w.

[8] J. Haegele, N. Panagiotopoulos, S. Cremers, J. Rahmer, J. Franke, R. L. Duschka, S. Vaalma, M. Heidenreich, J. Borgert, P. Borm, J. Barkhausen, and F. M. Vogt. Magnetic Particle Imaging: A Resovist Based Marking Technology for Guide Wires and Catheters for Vascular Interventions. *IEEE Transactions on Medical Imaging*, 35(10):2312–2318, 2016, doi:10.1109/TMI.2016.2559538.

[9] J. Haegele, S. Biederer, H. Wojtczyk, M. Gräser, T. Knopp, T. M. Buzug, J. Barkhausen, and F. M. Vogt. Toward cardiovascular interventions guided by magnetic particle imaging: First instrument characterization. *Magnetic Resonance in Medicine*, 69(6):1761–1767, 2013, doi:10.1002/mrm.24421.

[10] 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. *Journal of Endovascular Therapy*, 26(4):512–519, 2019, doi:10.1177/1526602819851202.

[11] J. Salamon, J. Dieckhoff, M. G. Kaul, C. Jung, G. Adam, M. Möddel, T. Knopp, S. Draack, F. Ludwig, and H. Ittrich. Visualization of spatial and temporal temperature distributions with magnetic particle imaging for liver tumor ablation therapy. *Scientific Reports*, 10(1):7480, 2020, doi:10.1038/s41598-020-64280-1.

[12] T. Knopp, T. F. Sattel, S. Biederer, J. Rahmer, J. Weizenecker, B. Gleich, J. Borgert, and T. M. Buzug. Model-Based Reconstruction for Magnetic Particle Imaging. *IEEE Transactions on Medical Imaging*, 29(1):12–18, 2010, doi:10.1109/TMI.2009.2021612.

[13] T. Knopp, S. Biederer, T. F. Sattel, J. Rahmer, J. Weizenecker, B. Gleich, J. Borgert, and T. M. Buzug. 2D model-based reconstruction for magnetic particle imaging. *Medical Physics*, 37(2):485–491, 2010, doi:10.1118/1.3271258.

[14] T. Kluth, P. Szwargulski, and T. Knopp. Towards accurate modeling of the multidimensional magnetic particle imaging physics. *New Journal of Physics*, 21(10):103032, 2019, doi:10.1088/1367-2630/ab4938.

[15] J. Rahmer, A. Halkola, B. Gleich, I. Schmale, and J. Borgert. First experimental evidence of the feasibility of multi-color magnetic particle imaging. *Physics in Medicine and Biology*, 60(5):1775–91, 2015, doi:10.1088/0031-9155/60/5/1775.

[16] C. Shasha, E. Teeman, K. M. Krishnan, P. Szwargulski, T. Knopp, and M. Möddel. Discriminating nanoparticle core size using multicontrast MPI. *Physics in Medicine & Biology*, 64(7):074001, 2019, doi:10.1088/1361-6560/ab0fc9.

[17] C. Stehning, B. Gleich, and J. Rahmer. Simultaneous magnetic particle imaging (MPI) and temperature mapping using multicolor MPI. *International Journal on Magnetic Particle Imaging*, 2(2), 2016, doi:10.18416/IJMPI.2016.1612001.

[18] M. Möddel, C. Meins, J. Dieckhoff, and T. Knopp. Viscosity quantification using multi-contrast magnetic particle imaging. *New Journal of Physics*, 20(8):083001, 2018, doi:10.1088/1367-2630/aad44b.

[19] M. Graeser, F. Thieben, P. Szwargulski, F. Werner, N. Gdaniec, M. Boberg, F. Griese, M. Möddel, P. Ludewig, D. van de Ven, O. M. Weber, O. Woywode, B. Gleich, and T. Knopp. Human-sized magnetic particle imaging for brain applications. *Nature Communications*, 10(1):1936, 2019, doi:10.1038/s41467-019-09704-x.

[20] T. Knopp and A. Weber. Sparse reconstruction of the magnetic particle imaging system matrix. *IEEE Transactions on Medical Imaging*, 32(8):1473–1480, 2013, doi:10.1109/TMI.2013.2258029.

[21] A. Weber and T. Knopp. Reconstruction of the Magnetic Particle Imaging System Matrix Using Symmetries and Compressed Sensing. *Advances in Mathematical Physics*, 2015, 2015, doi:10.1155/2015/460496.

[22] S. Ilbey, C. B. Top, A. Gungor, T. Cukur, E. U. Saritas, and H. E. Guven. Fast System Calibration with Coded Calibration Scenes for Magnetic Particle Imaging. *IEEE Transactions on Medical Imaging*, pp. 1–1, 2019, doi:10.1109/TMI.2019.2896289.

[23] A. von Gladiss, M. Ahlborg, T. Knopp, and T. M. Buzug. Compressed Sensing of the System Matrix and Sparse Reconstruction of the Particle Concentration in Magnetic Particle Imaging. *IEEE Transactions on Magnetics*, 51(2):1–4, 2015, doi:10.1109/TMAG.2014.2326432.

[24] M. Maass, M. Ahlborg, A. Bakenecker, F. Katzberg, H. Phan, T. M. Buzug, and A. Mertins. A Trajectory Study for Obtaining MPI System Matrices in a Compressed-Sensing Framework. *International Journal on Magnetic Particle Imaging*, 3(2), 2017, doi:10.18416/IJMPI.2017.1706005.

[25] M. Grosser, M. Moddel, and T. Knopp. Using Low-Rank Tensors for the Recovery of MPI System Matrices. *IEEE Transactions on Computational Imaging*, 6:1389–1402, 2020, doi:10.1109/TCI.2020.3024078.

[26] D. Donoho. Compressed sensing. *IEEE Transactions on Information Theory*, 52(4):1289–1306, 2006, doi:10.1109/TIT.2006.871582.

[27] M. Seeger, H. Nickisch, R. Pohmann, and B. Schölkopf. Optimization of k -space trajectories for compressed sensing by Bayesian experimental design. *Magnetic Resonance in Medicine*, 63(1):116–126, 2010, doi:10.1002/mrm.22180.

[28] F. Sherry, M. Benning, J. C. De los Reyes, M. J. Graves, G. Maierhofer, G. Williams, C.-B. Schonlieb, and M. J. Ehrhardt. Learning the Sampling Pattern for MRI. *IEEE Transactions on Medical Imaging*, 39(12):4310–4321, 2020, doi:10.1109/TMI.2020.3017353.

[29] E. Levine and B. Hargreaves. On-the-Fly Adaptive ${k}$ - Space Sampling for Linear MRI Reconstruction Using Moment-Based Spectral Analysis. *IEEE Transactions on Medical Imaging*, 37(2):557–567, 2018, doi:10.1109/TMI.2017.2766131.

[30] J. P. Haldar and D. Kim. OEDIPUS: An Experiment Design Framework for Sparsity-Constrained MRI. *IEEE Transactions on Medical Imaging*, 38(7):1545–1558, 2019, doi:10.1109/TMI.2019.2896180.

[31] Z. Ben-Haim and Y. C. Eldar. The Cramér-Rao Bound for Estimating a Sparse Parameter Vector. *IEEE Transactions on Signal Processing*, 58(6):3384–3389, 2010, doi:10.1109/TSP.2010.2045423.

[32] S. Liu, M. Fardad, E. Masazade, and P. K. Varshney. Optimal Periodic Sensor Scheduling in Networks of Dynamical Systems. *IEEE Transactions on Signal Processing*, 62(12):3055–3068, 2014, doi:10.1109/TSP.2014.2320455.

[33] Y. Zou and K. Chakrabarty. Sensor deployment and target localization in distributed sensor networks. *ACM Transactions on Embedded Computing Systems*, 3(1):61–91, 2004, doi:10.1145/972627.972631.

[34] A. Anis, A. Gadde, and A. Ortega. Efficient Sampling Set Selection for Bandlimited Graph Signals Using Graph Spectral Proxies. *IEEE Transactions on Signal Processing*, 64(14):3775–3789, 2016, doi:10.1109/TSP.2016.2546233.

[35] J. Swärd, F. Elvander, and A. Jakobsson. Designing sampling schemes for multi-dimensional data. *Signal Processing*, 150:1–10, 2018, doi:10.1016/j.sigpro.2018.03.011.

[36] S. Joshi and S. Boyd. Sensor Selection via Convex Optimization. *IEEE Transactions on Signal Processing*, 57(2):451–462, 2009, doi:10.1109/TSP.2008.2007095.

[37] S. Liu, S. P. Chepuri, M. Fardad, E. Masazade, G. Leus, and P. K. Varshney. Sensor Selection for Estimation with Correlated Measurement Noise. *IEEE Transactions on Signal Processing*, 64(13):3509–3522, 2016, doi:10.1109/TSP.2016.2550005.

[38] 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)*, 234–237, IEEE, 2021. doi:10.1109/ISBI48211.2021.9434109.

[39] 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. *Physics in Medicine and Biology*, 61(2):475–487, 2016, doi:10.1088/0031-9155/61/2/475.

[40] T. Goldstein and S. Osher. The Split Bregman Method for L1-Regularized Problems. *SIAM Journal on Imaging Sciences*, 2(2):323–343, 2009, doi:10.1137/080725891.

[41] F. Pukelsheim, Optimal design of experiments. Wiley, 1993, 454, ISBN: 047161971X 9780471619710.

[42] S. M. Kay, Fundamentals of Statistical Signal Processing: Estimation Theory. Prentice Hall PTR, 1993, 397–418.

[43] S. Reeves and Zhao Zhe. Sequential algorithms for observation selection. *IEEE Transactions on Signal Processing*, 47(1):123–132, 1999, doi:10.1109/78.738245.

[44] R. Broughton, I. Coope, P. Renaud, and R. Tappenden. Determinant and Exchange Algorithms for Observation Subset Selection. *IEEE Transactions on Image Processing*, 19(9):2437–2443, 2010, doi:10.1109/TIP.2010.2048150.

[45] W. W. Hager. Updating the Inverse of a Matrix. *SIAM Review*, 31(2):221–239, 1989, doi:10.1137/1031049.

[46] T. Knopp, P. Szwargulski, F. Griese, and M. Gräser. OpenMPIData: An initiative for freely accessible magnetic particle imaging data. *Data in Brief*, 28:104971, 2020, doi:10.1016/j.dib.2019.104971.

[47] Z. Wang, A. Bovik, H. Sheikh, and E. Simoncelli. Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004, doi:10.1109/TIP.2003.819861.

[48] 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. *Physics in Medicine and Biology*, 55(6):1577–1589, 2010, doi:10.1088/0031-9155/55/6/003.

[49] M. Ahlborg, C. Kaethner, T. Knopp, P. Szwargulski, and T. M. Buzug. Using data redundancy gained by patch overlaps to reduce truncation artifacts in magnetic particle imaging. *Physics in Medicine and Biology*, 61(12):4583–4598, 2016, doi:10.1088/0031-9155/61/12/4583.

[50] P. Szwargulski, M. Möddel, N. Gdaniec, and T. Knopp. Efficient Joint Image Reconstruction of Multi-Patch Data Reusing a Single System Matrix in Magnetic Particle Imaging. *IEEE Transactions on Medical Imaging*, 38(4):932–944, 2019, doi:10.1109/TMI.2018.2875829.

[51] T. Knopp, K. Them, M. Kaul, and N. Gdaniec. Joint reconstruction of non-overlapping magnetic particle imaging focus-field data. *Physics in Medicine and Biology*, 60(8):L15–L21, 2015, doi:10.1088/0031-9155/60/8/L15.

[52] J. Rahmer, J. Weizenecker, B. Gleich, and J. Borgert. Analysis of a 3-D System Function Measured for Magnetic Particle Imaging. *IEEE Transactions on Medical Imaging*, 31(6):1289–1299, 2012, doi:10.1109/TMI.2012.2188639.

[53] M. Maass and A. Mertins. On the Representation of Magnetic Particle Imaging in Fourier Space. *International Journal on Magnetic Particle Imaging*, 6(1), 2019, doi:10.18416/IJMPI.2019.1912001.

[54] G. Puy, P. Vandergheynst, and Y. Wiaux. On Variable Density Compressive Sampling. *IEEE Signal Processing Letters*, 18(10):595–598, 2011, doi:10.1109/LSP.2011.2163712.

[55] N. Gdaniec, H. Eggers, P. Boernert, M. Doneva, and A. Mertins, Novel Sampling Strategy for Abdominal Imaging with Incomplete Breathholds, in *Annual Meeting ISMRM*, 600, 2012.

[56] M. Boberg, T. Knopp, P. Szwargulski, and M. Moddel. Generalized MPI Multi-Patch Reconstruction using Clusters of similar System Matrices. *IEEE Transactions on Medical Imaging*, pp. 1–1, 2019, doi:10.1109/TMI.2019.2949171.

