International Journal on Magnetic Particle Imaging
Vol 8, No 2, Article ID 2212003, 7 Pages

Infinite Science Publishing

Research Article

System matrix compression using Chebyshev polynomials of first and second kind

Christine Droigk[a,*] · Marco Maass[a] · Alfred Mertins[a,b]

[a]Institute for Signal Processing, University of Lübeck, Lübeck, Germany
[b]German Research Center for Artificial Intelligence (DFKI), AI in Biomedical Signal Processing, Lübeck, Germany
*Corresponding author, email: c.droigk@uni-luebeck.de

Received 23 June 2022; Accepted 09 December 2022; Published online 23 December 2022

Abstract

A common procedure for the reconstruction in Lissajous-type magnetic particle imaging is solving a linear system of equations which is based on the measurement of the so-called system matrix and the induced voltage signal of the unknown magnetic particle distribution. To speed up the reconstruction process and to reduce memory consumption, different compression techniques for the system matrix have been investigated. In this work, we propose a system matrix compression using the tensor product of Chebyshev polynomials of first and second kind. This method is motivated by recently published theoretical findings on the system function. For evaluation, simulated and real-world system matrices have been compressed with the proposed method and other state-of-the-art techniques. When using only one compression coefficient per row, the proposed compression outperforms the compared methods for all tested system matrices in terms of the normalized error metric and better reconstruction results.

I. Introduction

Magnetic particle imaging (MPI) is a tomographic imaging technique that can show the distribution of superparamagnetic iron-oxide nanoparticles (SPIONs) inside a field of view (FOV) [1]. Applying different magnetic fields, a field-free point (FFP) is generated that moves along a trajectory through the FOV. The SPIONs near the FFP change their magnetization and induce a voltage signal in the receive coils of the MPI scanner which is used to reconstruct the SPION distribution. In case of a FFP Lissajous trajectory, the reconstruction of the SPION distribution is typically done by solving a regularized system of linear equations. For this purpose, the system matrix that links the induced voltage signal to the SPION distribution is necessary. As the system matrix can be very large and memory-demanding, compression techniques have been proposed [2–4]. Since the structure of the system matrix shows similarities to the tensor product of Chebyshev polynomials (CPs) of second kind, the discrete Chebyshev transform (DTT) has been used for system matrix compression [2, 5]. In [5] it was shown that for few coefficients, the DTT achieved better compression results on the drive-field FOV (DF-FOV) than the discrete cosine transform (DCT). Recently, in [6] a mathematical formulation of the equilibrium model for Lissajous FFP trajectory based MPI has been provided that shows the relationship between tensor products of CPs and the system function. Analysis of the provided relationship leads to the assumption that the system function might be approximated better by the tensor product of CPs of first and second kind. Based on this assumption we propose a system matrix compression that uses both CPs of first and second kind. To evaluate the ability to compress, experiments are performed on both differently simulated system matrices and measured system

matrices. The DCT, DTT and the proposed method are used to compress the DF-FOV, respectively. Also a reconstruction with the real-world system matrix of the OpenMPIData [7] as well as its different compressions is performed. It is found that the proposed compression achieves the best results on the simulations based on the Langevin model and outperforms the DTT and also the DCT when three or less compression coefficients are used. The more realistic the simulation model becomes, the smaller the advantage is. However, even on real data, an advantage remains with strong compression for the proposed method with CPs of the first and second kind, which is also shown in better reconstruction results.

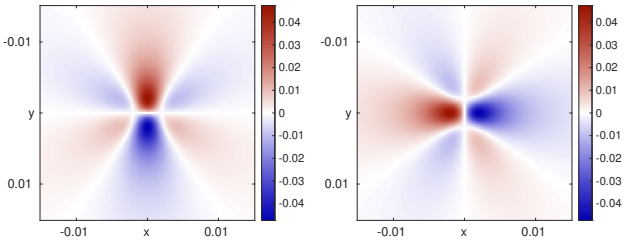

Figure 1: First (left) and second (right) component of the kernel $\frac{\partial^2}{\partial z_1 \partial z_2}\mathscr{L}(\beta \boldsymbol{G}\boldsymbol{z})$, corresponding to the x- and y-coils. Here, $\beta = 1.2\cdot 10^{-3}\,\mathrm{mA}^{-1}$ and $G_x = G_y = 1\mu_0^{-1}\,\mathrm{Tm}^{-1}$ with μ_0 the vacuum permeability are chosen. One can see the similarity to discrete differentiation operators in the y- and x-direction, respectively.

II. Methods and materials

II.I. System function

In the common MPI setup, the voltage signal induced by the SPIONs is measured in L orthogonal receive coils. Its relation to the SPION distribution in the temporal Fourier domain is

$$\boldsymbol{u}_k = \int_\Omega \boldsymbol{s}_k(\boldsymbol{r}) c(\boldsymbol{r})\,\mathrm{d}\boldsymbol{r}, \tag{1}$$

where $\boldsymbol{u}_k \in \mathbb{C}^L$ is the k-th frequency component of the voltage signal, $\boldsymbol{s}_k : \mathbb{R}^L \to \mathbb{C}^L$ denotes the k-th frequency component of the system function, $c : \mathbb{R}^L \to \mathbb{R}$ is the particle distribution, and $\Omega \subset \mathbb{R}^L$ the FOV.

In the following, we restrict us to the case $L = 2$, but the same procedure can also be applied for $L = 3$. In [6] it was derived that the system function components for a two-dimensional Lissajous FFP-trajectory in a simplified physical model can be expressed as

$$\begin{aligned}\boldsymbol{s}_k(\boldsymbol{r}) = \sum_{\lambda\in\mathbb{Z}} \boldsymbol{C}_{k,\lambda} \int_{\mathbb{R}^2} \left[\frac{\partial^2}{\partial z_1 \partial z_2}\mathscr{L}(\beta \boldsymbol{G}\boldsymbol{z})\right]_{\boldsymbol{z}=\boldsymbol{r}-\boldsymbol{u}} \\ V_{-k+\lambda N_B}\left(\frac{G_x}{A_x}u_1\right) V_{k-\lambda(N_B-1)}\left(\frac{G_y}{A_y}u_2\right)\mathrm{d}\boldsymbol{u}\end{aligned} \tag{2}$$

with a known, constant matrix $\boldsymbol{C}_{k,\lambda}$, $\boldsymbol{A} = \mathrm{diag}(A_x, A_y)$ denoting the drive-field amplitudes in x- and y-direction, and $\boldsymbol{G} = \mathrm{diag}(G_x, G_y)$ the applied gradients of the selection field. For details of $\boldsymbol{C}_{k,\lambda}$ we refer to [6]. Besides,

$$V_n(x) = \begin{cases} \mathrm{rect}\left(\frac{x}{2}\right)\left(-\frac{U_{|n|-1}(x)\sqrt{1-x^2}}{|n|}\right), & n \neq 0 \\ \frac{\pi}{2}\mathrm{sgn}(x+1) - \mathrm{rect}\left(\frac{x}{2}\right)\arccos(x), & n = 0, \end{cases} \tag{3}$$

where U_n denotes the CP of second kind and n-th order, and the Langevin function $\mathscr{L} : \mathbb{R} \to \mathbb{R}$ describes the particle magnetization in the equilibrium model. Its multi-dimensional extension $\mathscr{L} : \mathbb{R}^L \to \mathbb{R}^L$ is defined as

$$\mathscr{L}(\boldsymbol{x}) = \begin{cases} \mathscr{L}(\|\boldsymbol{x}\|)\frac{\boldsymbol{x}}{\|\boldsymbol{x}\|}, & \|\boldsymbol{x}\| \neq 0 \\ \boldsymbol{0}, & \|\boldsymbol{x}\| = 0. \end{cases} \tag{4}$$

Thus, the system function consists of tensor products of weighted CPs of the second kind, but those are subsequently convolved with the kernel $\frac{\partial^2}{\partial z_1 \partial z_2}\mathscr{L}(\beta \boldsymbol{G}\boldsymbol{z})$. The kernel is shown in Figure 1. As outlined in [8], one can see that it forms an approximate derivative in one direction, namely the first component in the y-direction, and the second component in the direction of x. However, applying the product rule on the derivative of V_n for $n \neq 0$ one gets

$$\frac{\mathrm{d}}{\mathrm{d}x}\frac{-U_{|n|-1}(x)\sqrt{1-x^2}}{|n|} = \frac{T_{|n|}(x)}{\sqrt{1-x^2}}, \quad x \in [-1,1]. \tag{5}$$

Here, T_n is the CP of first kind and order n. Besides, it yields

$$\frac{\mathrm{d}}{\mathrm{d}x}V_0(x) = \frac{T_0(x)}{\sqrt{1-x^2}}, \quad x \in [-1,1]. \tag{6}$$

This means that the system function can be approximated as the tensor product of CPs of first and second kind. The direction here is different for the x- and y-coil.

II.II. System matrix compression

Applying the above derived insight to the procedure of [5], a modified system matrix compression is quickly obtained: For compression of a system matrix describing an $N \times N$ FOV, the transformation matrix $\boldsymbol{B} \in \mathbb{R}^{N^2 \times N^2}$ must be set up. For an orthogonal transformation, the coefficient matrix $\hat{\boldsymbol{S}} \in \mathbb{C}^{M \times N^2}$ can then be obtained via $\hat{\boldsymbol{S}} = \boldsymbol{S}\boldsymbol{B}$ with $\boldsymbol{S} \in \mathbb{C}^{M \times N^2}$ denoting the original system matrix. If $\boldsymbol{B}$ is unitary it yields $\boldsymbol{S} = \hat{\boldsymbol{S}}\boldsymbol{B}^{\mathrm{H}}$. To obtain a unitary transformation matrix and thus to improve calculation efficiency, like in [5], the $N \in \mathbb{N}$ grid points

$$\boldsymbol{r}^{\mathrm{Cheb}} = \cos\left(\frac{(n+0.5)\pi}{N}\right), \quad n = 0, \ldots, N-1 \tag{7}$$

10.18416/ijmpi.2022.2212003

and the weighting

$$w_m^U = \begin{cases} \sqrt{\frac{2}{N}} & m = 0, \dots, N-2 \\ \sqrt{\frac{1}{N}} & m = N-1 \end{cases} \tag{8}$$

$$w_m^T = \begin{cases} \sqrt{\frac{1}{N}} & m = 0 \\ \sqrt{\frac{2}{N}} & m = 1, \dots, N-1 \end{cases} \tag{9}$$

can be used [9]. For $x, y \in [-1, 1]$, we define

$$\mathrm{UT}_{k,l}(x, y) = \sqrt{1-x^2} U_k(x) T_l(y) \tag{10}$$

and

$$\mathrm{TU}_{k,l}(x, y) = T_k(x) U_l(y) \sqrt{1-y^2}. \tag{11}$$

Then, obtaining k and l from the relationship $m = k + lN$, the transformation matrices can be written as

$$\boldsymbol{B}_{\mathrm{UT}} = (w_k^U \, w_l^T \, \mathrm{UT}_{k,l}(\boldsymbol{r}_n^{\mathrm{Cheb}}))_{n=0,\dots,N^2-1; m=0,\dots,N^2-1} \tag{12}$$

and

$$\boldsymbol{B}_{\mathrm{TU}} = (w_k^T \, w_l^U \, \mathrm{TU}_{k,l}(\boldsymbol{r}_n^{\mathrm{Cheb}}))_{n=0,\dots,N^2-1; m=0,\dots,N^2-1}, \tag{13}$$

where the former is used for compressing the x-receive path and the latter for compressing the y-receive path. Besides, $\boldsymbol{r}_n^{\mathrm{Cheb}}$ denotes the n-th entry of the tensor product grid constructed with the one-dimensional grid points in (7).

Since V_n in (3) for $n \neq 0$ is nonzero only within the DF-FOV, it is reasonable to apply the compression as in [5] only to the system matrix in the DF-FOV. To obtain an appropriate compression, a large fraction of the obtained coefficients in $\hat{\boldsymbol{S}}$ must be discarded. This can be done in form of global [2] or local thresholding [3, 5].

II.III. Experiments

To evaluate the proposed system matrix compression, different two-dimensional system matrices and compression techniques were tested. The proposed compression has been performed using the transformation matrix B_{UT} for the x-receive path of the system matrix and B_{TU} for the y-receive path. It is called DTT-UT compression in the following. For comparison purposes, a compression has also been performed using only CPs of second kind as done in [5]. This type of compression is called DTT-UU in the following. Besides, the discrete cosine transform of type II has been applied which is referred to as DCT in the following. The DCT compression is performed on an equidistant grid while the DTT-UU and the proposed DTT-UT compressions are performed on a Chebyshev grid. All compression techniques are restricted on the DF-FOV for a better comparison. For the compression, we specify a number of allowed coefficients per row, i.e., per frequency component, as was done in [5]. This corresponds to a separate local threshold per frequency component which has shown advantages over a global threshold [3]. Because the presented theory relies on the Langevin model, a system matrix based on this physical model has been simulated and used for compression. Besides, a more realistic simulation has been carried out. For this, the physical model described by both [10] and [11] has been used. For an anisotropy gradient of $g_{K_{\mathrm{anis}}} = 1250\,\mathrm{Jm}^{-3}$, the simulated system matrices are close to real-world system matrices. All simulated system matrices had a size of 41×41 pixels. Lastly, the two-dimensional system matrix of the OpenMPIData [7] has been compressed. The provided system matrix has been background corrected by subtraction of the provided empty measurement. Besides, the measured system matrix is not perfectly centered. This was corrected by a translation to obtain a better centering. Because there is an overscan area included in the measurement, the system matrix has been cropped so that it consisted of the DF-FOV only, and then interpolated on a finer grid of size 41×41 using bicubic splines. A frequency selection has been performed by discarding rows with a signal-to-noise ratio below one. Since the matrix is given on a Cartesian grid, a bicubic spline interpolation onto a Chebyshev grid has been performed after the cropping. To compare the two compression methods, the compressed coefficient matrix was transformed back. For a later reconstruction the error on a system matrix with equidistant grid is more relevant. Therefore the compressed SMs were interpolated on the Cartesian grid by cubic spline interpolation and then compared with the original matrix, i.e., its deviation from the original system matrix was calculated. Two different error metrics were used for this purpose. Firstly, we consider the error metric used by [5] which is given as

$$\sigma^\Gamma = \frac{1}{|K|} \sum_{k \in K} \frac{\|\boldsymbol{S}_k - \boldsymbol{S}_k^\Gamma\|_2}{\|\boldsymbol{S}_k\|_2}. \tag{14}$$

Here, K is a set of frequency components and $\boldsymbol{S}_k$ denotes the k-th row of the system matrix $\boldsymbol{S}$. Besides, $\boldsymbol{S}^\Gamma$ denotes a compressed system matrix using Γ coefficients per row. As a second error metric we consider the following

$$\rho^\Gamma = \frac{1}{|K|} \sum_{k \in K} \left\| \frac{\boldsymbol{S}_k}{S_{k,\max}} - \frac{\boldsymbol{S}_k^\Gamma}{S_{k,\max}^\Gamma} \right\|_2 \tag{15}$$

where $S_{k,\max}$ denotes the maximum value of $|\boldsymbol{S}_k|$. To prevent false extremely high values for $S_{k,\max}$ due to noise artifacts, the upper value of the 99 % percentile of $|S_k|$ is chosen here for $S_{k,\max}$. The error metric ρ^Γ is better suited to show the structural similarity because differences in the magnitude are considered less. For reconstruction, the structural similarity of the system matrices might be more relevant to prevent reconstruction artifacts, while a similar magnitude might be more relevant for obtaining the right concentration value.

For a better understanding of the differences of these two metrics, the energies of the original and the differ-

ently compressed system matrices are calculated per row and compared.

As argued in Section II.II, it is reasonable to apply the compression only to the DF-FOV. But the compression of the complete FOV including an overscan area is of interest because the inclusion of an overscan region can reduce artifacts [12]. Therefore, we also test the DTT-UT compression on the OpenMPIData system matrix including the overscan region and compute both error metrics σ^Γ and ρ^Γ. However, as neither the CPs of first nor second kind fit the underlying model when considering an overscan region, no advantage of the proposed DTT-UT compression method can be expected here.

Lastly, we show some reconstruction results that were obtained by the compressed OpenMPIData system matrices on the DF-FOV. We reconstruct the resolution phantom using the compressed system matrices with $\Gamma = 1$ and $\Gamma = 6$, and compare the root mean square errors of the obtained reconstructions with respect to the reconstruction with the uncompressed real-world system matrix. To focus on structural similarity, all reconstructions were normalized beforehand by division through the maximum value. For the reconstruction, the standard procedure was performed, i.e., the discrete minimization problem

$$\underset{\boldsymbol{c}\in\mathbb{R}^N,\boldsymbol{c}\geq 0}{\arg\min} \|\boldsymbol{S}\boldsymbol{c}-\boldsymbol{u}\|_2^2+\lambda\|\boldsymbol{c}\|_2^2 \tag{16}$$

has been solved using the Kaczmarz algorithm. The regularization parameter λ has been optimized for each reconstruction.

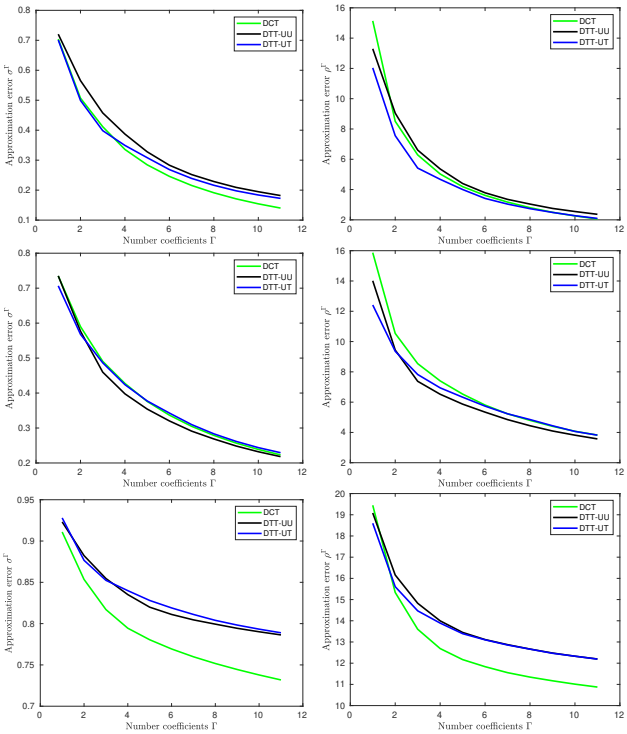

Figure 2: Error metrics σ^Γ (left) and ρ^Γ (right) dependent on the number of used coefficients for the compression of different system matrices. Top: Langevin model simulation. Middle: Simulation including anisotropy effects. Bottom: Real-world OpenMPIData.

III. Results

The errors of the proposed DTT-UT and the DTT-UU and DCT baseline can be seen in Figure 2. As expected, the proposed compression method outperforms the DTT-UU compression in both error metrics for the system matrix relying on the Langevin model. Compared with the DCT, the DTT-UT compression is slightly better for few coefficients in terms of the σ^Γ error, and gets worse when more coefficients are used. When comparing the ρ^Γ metric of these compression techniques, the DTT-UT also outperforms the DCT and obtains the best results.

This behavior changes when the system matrices deviate from the Langevin model. But still the DTT-UT outperforms the other methods when only one or two coefficients per row are used for compression. Using more coefficients, the DTT-UT gets worse than the DTT-UU, but performs similar to the DCT.

These observations change again when compressing the real-world system matrices. Here, the DCT outperforms the other compressions significantly when looking at the σ^Γ error. Comparing the error ρ^Γ, the proposed DTT-UT compression obtains the best result when using one coefficient only per row. Using more coefficients, the DTT-UT is better or equal compared to the DTT-UU, but the DCT outperforms both in these cases.

A comparison of the original system matrix and the different compressions for $\Gamma = 1$ can be seen in Figure 3. The frequency components $k = 115$ and $k = 213$ are shown exemplarily. One can see the different shapes of each compression method. The position of the wave hills from the proposed DTT-UT compression seems to match the actual position slightly better than the two other methods. Instead the obtained magnitude of the wave hills is the lowest and differs the most from the original system matrix. The magnitude of the DTT-UU method is close to the DTT-UT, and the DCT compression has the highest values, though it is still away from the original by about a factor of two. This seems to be the reason why the error metrics σ^Γ and ρ^Γ behave differently. These differences are investigated in Figure 4 where the energy distribution of the system matrix rows is shown. For nearly all frequency components there is the same pattern: The original system matrix has the highest energy followed by the DCT compression. The DTT-UU and DTT-UT compressions have a lower, similar energy.

The error curves of the OpenMPIData system matrix compression including an overscan region are shown in

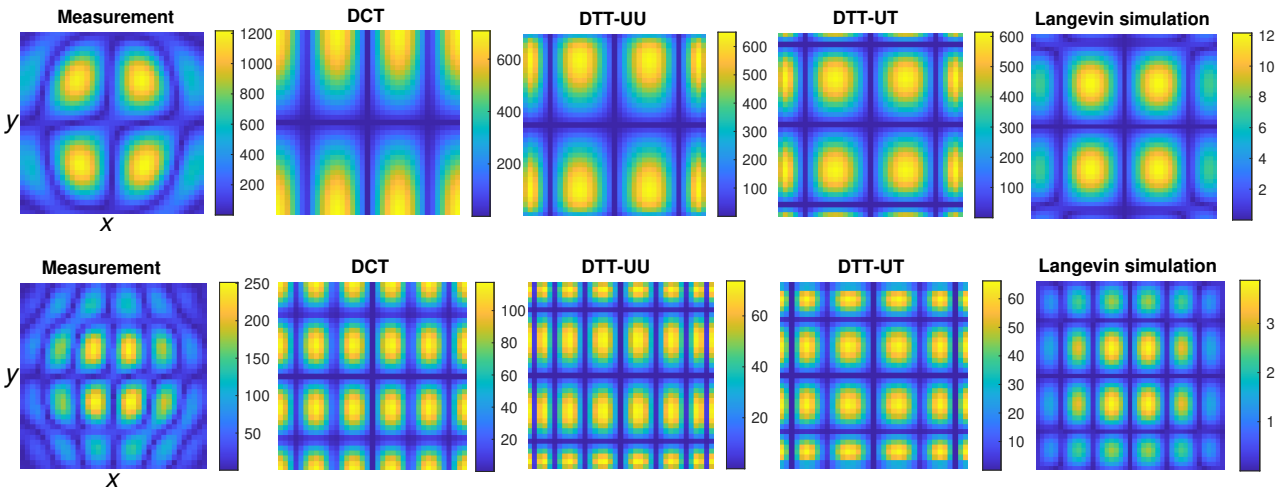

Figure 3: Original system matrix component and its compressed variants for the three compression methods and $\Gamma = 1$. Two frequency components are shown exemplarily. Top: The frequency component $k = 115$ of the x-receive path. Bottom: The frequency component $k = 213$ of the x-receive path. The simulated system matrix components (right) are shown for comparison. For the x-receive path shown here, the Chebyshev polynomials of first kind are used in the y-dimension for the proposed DTT-UT compression. In this dimension the position of the maxima and minima is better matched by the proposed DTT-UT compression than by the other methods.

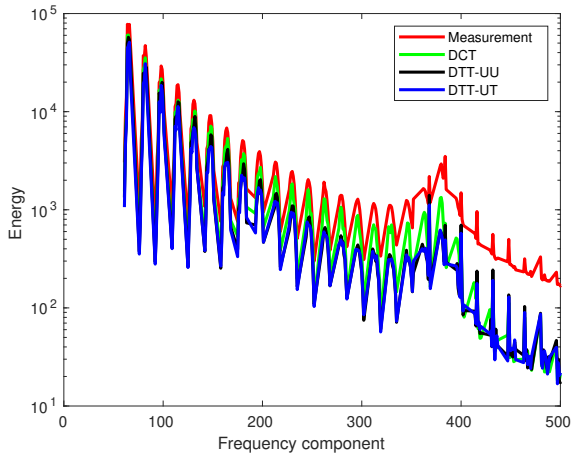

Figure 4: Energy of the OpenMPIData system matrices and the obtained compressions with $\Gamma = 1$. The lower energy of all compressed system matrices is visible.

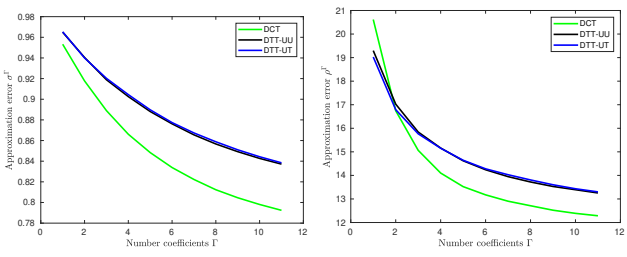

Figure 5: Error metrics σ^Γ (left) and ρ^Γ (right) dependent on the number of used coefficients Γ for the OpenMPIData system matrix including the overscan region. As expected, the DTT-UU and DTT-UT compression methods achieve similar results.

Figure 5. In terms of the error metric σ^Γ, which does not compensate different amplitudes, the DCT outperforms the two other methods. When considering the other error metric ρ^Γ the proposed DTT-UT method outperforms the DCT for one compression coefficient, and is only slightly better than the DTT-UU method. For two compression coefficients, all methods achieve similar results, and for three or more used coefficients, the DCT obtains the best compression. Thus, when compressing measured data, the advantages and disadvantages over the DCT are similar whether compressing the DF-FOV only or the FOV including an overscan region. The main difference lies in the relationship between the DTT-UT and the DTT-UU method: The advantage of the proposed DTT-UT over the DTT-UU method shrinks when including the overscan region, as it was expected.

The reconstruction results for the resolution phantom are shown in Figures 6 and 7. Note that only the DF-FOV is reconstructed. For a compression with only one coefficient $\Gamma = 1$ per row, the proposed DTT-UT compression obtains the best RMSE, followed by the DTT-UU and DCT compression. This corresponds to the error ρ^Γ of the compressed system matrices, where for $\Gamma = 1$ the same order is obtained. The better RMSE in the reconstructed image is also visible in terms of less background artifacts and the better detailed reconstruction of the phantom's upper arm. Looking at the compression obtained with $\Gamma = 6$ coefficients per row, the DCT compression achieves the best reconstruction, while the DTT-UU and DTT-UT reconstructions are very similar. This also corresponds to the profile of the error curve of ρ^Γ in Figure 2.

10.18416/ijmpi.2022.2212003

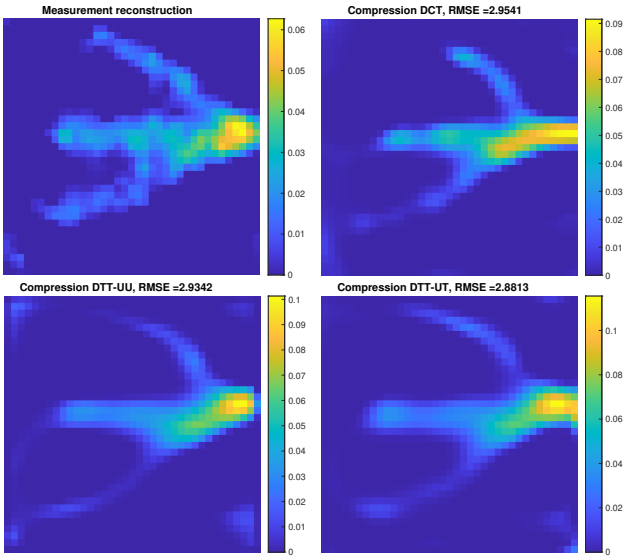

Figure 6: Reconstructions of the resolution phantom with $\Gamma = 1$. In the upper left corner, the reconstruction obtained by the uncompressed real-world system matrix is shown. The used compression method and the obtained RMSE is stated above the reconstructed images.

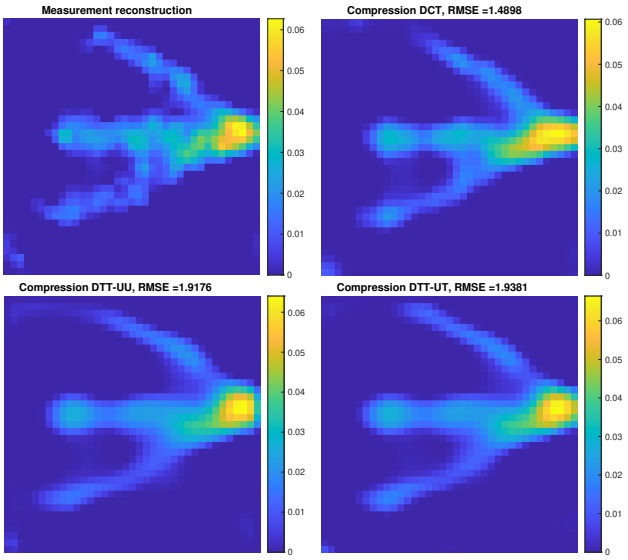

Figure 7: Reconstructions of the resolution phantom with $\Gamma = 6$. In the upper left corner, the reconstruction obtained by the uncompressed real-world system matrix is shown. The used compression method and the obtained RMSE is stated above the reconstructed images.

IV. Discussion

The consideration of the proposed DTT-UT compression originates from a simplified model and outperforms the compared compression methods on this model. This shows that the observations made in the derivation are meaningful. The further the system matrix deviates from this ideal, the more the advantage over the other methods shrinks. Though, when considering the real-world system matrices, the proposed DTT-UT compression still outperforms the DTT-UU and the DCT compression when using one compression coefficient per row only in terms of a lower error metric ρ^1 and a better RMSE of the obtained reconstruction image. This also shows that the system function of Lissajous-type MPI actually is more similar to a tensor product of CPs of first and second kind instead of the long assumed tensor product of CPs of second kind. This is also reflected in the comparison of the system matrix components in Figure 3, in which the structure of the DTT-UT compression comes closest to both the simulated and the original real-world system matrix. Though, when using more than one coefficient per row for compression, the DCT obtains the best results on real data. Regardless, balancing of the energy or maximum amount of the rows of the compressed system matrices should take place so that the reconstructed images match those of the measurement in the amount of concentration.

However, as only the DF-FOV is compressed, no overscan region can be included which has been shown to reduce reconstruction artifacts [12]. Though, in the provided reconstructions using the DF-FOV only, no significant reconstructions artifacts could be observed, because most of the tracer material was located inside the DF-FOV.

Although the physical model is not fitted by neither CPs of first or second kind when including an overscan region, we found that the proposed DTT-UT compression can also be used without quality loss compared to the DTT-UU method. As both DTT-UU and DTT-UT outperform the DCT when using only one compression coefficient in terms of a lower ρ^Γ metric, the wave hill pattern seems to be matched better in this case. However, when compressing real data with three or more coefficients per row, the DCT seems to be still the method of choice.

V. Conclusion

Based on the mathematical description of the simplified physical MPI model without relaxation effects, it was found that the system function in Lissajous-type MPI is more similar to the tensor product of CPs of the first kind with CPs of the second kind, unlike what was previously thought. Based on this consideration, a compression of the system matrix was proposed using as basis functions the products of CPs of first and second kind. The proposed compression was compared with the compression using only CPs of second kind and the DCT compression on different physical simulations and on real-world data. The closer the system matrix is to the simplified model, the better the proposed compression method

performs compared to the other methods. When using only one compression coefficient, the proposed compression outperforms the other methods for all tested system matrices including the real-world system matrices in terms of the normalized error metric ρ^{Γ}. When three or more coefficients are used, the proposed DTT-UT method performs still better on the Langevin model, but is outperformed on the OpenMPIData by the DCT.

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. Lampe, C. Bassoy, J. Rahmer, J. Weizenecker, H. Voss, B. Gleich, and J. Borgert. Fast reconstruction in magnetic particle imaging. *Physics in Medicine & Biology*, 57(4):1113–1134, 2012, doi:10.1088/0031-9155/57/4/1113.

[3] T. Knopp and A. Weber. Local System Matrix Compression for Efficient Reconstruction in Magnetic Particle Imaging. *Advances in Mathematical Physics*, 2015:1–7, 2015, doi:10.1155/2015/472818.

[4] M. Maass, K. Bente, M. Ahlborg, H. Medimagh, H. Phan, T. M. Buzug, and A. Mertins. Optimized Compression of MPI System Matrices Using a Symmetry-Preserving Secondary Orthogonal Transform. *International Journal on Magnetic Particle Imaging*, 2(1), 2016, doi:10.18416/IJMPI.2016.1607002.

[5] L. Schmiester, M. Möddel, W. Erb, and T. Knopp. Direct Image Reconstruction of Lissajous-Type Magnetic Particle Imaging Data Using Chebyshev-Based Matrix Compression. *IEEE Transactions on Computational Imaging*, 3(4):671–681, 2017, doi:10.1109/TCI.2017.2706058.

[6] 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.

[7] 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.

[8] C. Droigk, M. Maass, and A. Mertins. Direct multi-dimensional Chebyshev polynomial based reconstruction for magnetic particle imaging. *Physics in Medicine & Biology*, 67(4):045014, 2022, doi:10.1088/1361-6560/ac4c2e.

[9] J. Mason and D. C. Handscomb, Chebyshev Polynomials. Chapman and Hall/CRC, 2002, doi:10.1201/9781420036114.

[10] H. Albers and T. Kluth. Immobilized nanoparticles with uniaxial anisotropy in multi-dimensional Lissajous-type excitation: An equilibrium model approach. *International Journal on Magnetic Particle Imaging*, 8(1 Suppl. 1), 2022, doi:10.18416/IJMPI.2022.2203048.

[11] M. Maass, C. Droigk, M. Eulers, and A. Mertins. An analytical equilibrium solution to the Néel relaxation Fokker-Planck equation. *International Journal on Magnetic Particle Imaging*, 8(1 Suppl. 1), 2022, doi:10.18416/IJMPI.2022.2203008.

[12] 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 & Biology*, 61(2):475–487, 2016, doi:10.1088/0031-9155/61/2/475.