International Journal on Magnetic Particle Imaging
Vol 9, No 1, Suppl 1, Article ID 2303003, 4 Pages

Infinite Science Publishing

Proceedings Article

Exploiting the Fourier Neural Operator for faster magnetization model evaluations based on the Fokker-Planck equation

Tobias Knopp[a,b] · Hannes Albers[c] · Mirco Grosser[a,b] · Martin Möddel[a,b] · Tobias Kluth[c]

[a]Section for Biomedical Imaging, University Medical Center Hamburg-Eppendorf, Hamburg, Germany
[b]Institute for Biomedical Imaging, Hamburg University of Technology, Hamburg, Germany
[c]Center for Industrial Mathematics, University of Bremen, Bremen, Germany
*Corresponding author, email: t.knopp@uke.de

Abstract

Accurate modeling of the mean magnetic moment of an ensemble of magnetic particles in dynamic magnetic fields is a challenging task that requires sophisticated differential equation solvers. However, these methods are computationally costly and therefore not practical for long excitation sequences such as those of the Lissajous type. In this paper we propose to accelerate simulations by using a neural network mapping from the input parameter functions that are applied to the original particle simulator directly to the mean magnetic moment output function. The architecture of the neural network is based on the Fourier neural operator, which allows to train mappings between function spaces. Our results show that the particle simulation can be accelerated by a factor of about 200 while the relative error of the neural network simulator remains below 1.5 %.

I. Introduction

Accurate knowledge of the mean magnetic moment of magnetic nanoparticles (MNP) is crucial for solving the image reconstruction problem in magnetic particle imaging (MPI). While time-consuming calibration methods are still commonly used, model-based approaches have recently gained substantially in accuracy [1]. These methods require the solution of a partial differential equation (PDE) like the Fokker-Planck equation [2–4]. Using the methods of lines, the PDE can be transformed into an ordinary differential equation (ODE). Solving this ODE can be computationally costly , especially for Lissajous type imaging sequences. For instance, the system matrix of a 2D Lissajous type sequence evaluated on a 30 × 30 grid requires about 64 min of simulation time on a single CPU core, whereas a 3D system matrix evaluated on a 30 × 30 × 30 grid requires 40 d. This is much longer than the robot-based calibration, which would require just about 20 h. These numbers hold for a single set of parameters describing the nanoparticles' properties, which need to be known prior to simulation. If the parameters are unknown, several such simulations are required to solve the parameter identification problem [5, 6].

One way to accelerate MNP simulation is to exploit parallel computing, which allows speedups in the range of 100–1000. An alternative that we use in this work is to exploit machine learning for the solution of PDEs. In particular, we take the state of the art Fourier Neural Operator (FNO) approach [7, 8], which learns the mapping from the PDE input parameters to its solutions and develop a framework for computing the mean magnetic moment of MNP induced by 3D Lissajous-type imaging sequences evaluated on 3D grids in a shorter time than an actual calibration would take.

10.18416/ijmpi.2023.2303003

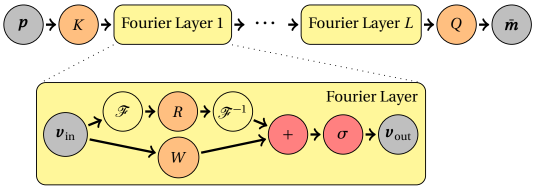

Figure 1: Architecture of the FNO. K and Q are dense layers that adapt the channel dimensions so that the Fourier layers operate on N_{C} channels. Within each Fourier layer, the data path is split into two processing paths, of which one applies a filter R in Fourier domain and the other applies a weighting plus channel combination W in time domain.

II. Methods and materials

Particle magnetization model: For the simulation of the mean magnetic moment of MNP we follow the Fokker-Planck equation approach for the Landau-Lifshitz-Gilbert equation as already described in [1, 9]. Without loss of generality, we consider the Néel relaxation case, which can approximate both the case of immobilized [6] and free particles suspended in fluid [1] sufficiently well. The solution of a Fokker-Planck simulator depends on the following input parameters:

1. the applied magnetic field $\boldsymbol{H} : [0, T] \to \mathbb{R}^3$ where T is the length of the considered time interval
2. the particle diameter $D \in \mathbb{R}_+$,
3. the anisotropy constant $K_{\mathrm{anis}} \in \mathbb{R}_+$,
4. the easy axis $\boldsymbol{n} \in \mathbb{S}^2$.

We may combine the last two parameters into $\boldsymbol{K}_{\mathrm{anis}} = K_{\mathrm{anis}}\boldsymbol{n} \in \mathbb{R}^3$. These parameters define a function $\boldsymbol{p} : [0, T] \to \mathbb{R}^{N_p}$, $\boldsymbol{p}(t) = (\boldsymbol{H}(t), D, \boldsymbol{K}_{\mathrm{anis}})$ with $N_p = 7$. The output of the Fokker-Planck simulator is the mean magnetic moment $\bar{\boldsymbol{m}} : [0, T] \to \mathbb{R}^3$. The solver $F_{\text{Fokker-Plank}}$ can thus be considered to be an operator that maps from the function $\boldsymbol{p}$ to the function $\bar{\boldsymbol{m}}$, i.e. $F_{\text{Fokker-Plank}}(\boldsymbol{p}) = \bar{\boldsymbol{m}}$, which comprises PDE solution and expectation operator.

Fourier neural operator: The basic idea of the neural operator framework [7] is to train a (deep) neural network, which operates on functions instead of discrete arrays. This is achieved by transforming the domain from e.g. the discrete time domain into the Fourier domain, which is then independent of the resolution of the discretization in time domain. When using a Fourier transform, the operator is named Fourier neural operator (FNO) [8]. More specifically, it consists of several so-called Fourier layers, which apply a discrete Fourier transform to the input matrix along the time dimension and apply both a low pass and a learned filter to the Fourier coefficients as shown in Fig. 1. All filters are implemented as dense layers, which also allow for channel combination. FNOs allow to approximate solutions to an entire class of differential equations with high accuracy, since the FNO gets the parameters of the differential equation as input. I.e. they approximate the abstract operator which maps a specific parameter to the corresponding solution.

To apply the FNO for MNP simulation, we replace the operator $F_{\text{Fokker-Plank}}$ by the operator F_{FNO} operating on the same parameter function $\boldsymbol{p}$. In the actual implementation, the parameter function $\boldsymbol{p}$ is discretized along the time dimension at $N_t = 200$ time points $t_k = \frac{kT}{N_t}$, $k = 1, 2, \ldots, N_t$, such that the input of the FNO is the matrix $\boldsymbol{p}_{\mathrm{S}} = \left((\boldsymbol{p}(t_k))_d\right)_{k=1,\ldots,N_t; d=1,\ldots,N_p} \in \mathbb{R}^{N_t \times N_p}$. The output of the FNO is a discrete sequence of mean magnetic moments $\boldsymbol{m}_{\mathrm{S}} \in \mathbb{R}^{N_t \times 3}$.

Data generation and training: For network training and testing, $N_{\mathrm{data}} = 5000$ data tuples $(\boldsymbol{p}_{\mathrm{S}}^{(i)}, \bar{\boldsymbol{m}}_{\mathrm{S}}^{(i)})$, $i = 1, \ldots, N_{\mathrm{data}}$ are generated (90 % training, 10 % testing). The time dependent field $\boldsymbol{H}$ is generated using N_t uniformly distributed random numbers within the interval $[-H_{\max}, H_{\max}]$ for each component x, y, z with $H_{\max} = 20\ \mathrm{mT}/\mu_0$. The sampling rate was chosen to be 2.5 MHz resulting in a time length of 80 μs. As we expect magnetic fields with limited slope $\boldsymbol{H}$, each component of the magnetic field vector is convolved in the time dimension with a Gaussian kernel with randomly chosen width σ from the interval $[4, 20]$. $\boldsymbol{n} \in \mathbb{S}^2$ and $K_{\mathrm{anis}} \in (0, 1600]\ \mathrm{Jm}^{-3}$ are again drawn from uniform distributions. The particle core diameter is chosen to be constant: $D = 20$ nm. The ground truth solution for the mean magnetic moment $\bar{\boldsymbol{m}}_{\mathrm{S}}$ is generated by Fokker-Planck simulation. Network training is performed by minimizing a relative ℓ^2-loss using Adam. The network is trained with decaying learning rate $\eta \in \{10^{-3}, 10^{-4}, 10^{-5}\}$, using 100 epochs for each rate. After training, the relative error is 0.828 % on the training data and 0.959 % on independently generated test data.

We trained the network using functions defined on the short and constant time interval $[0, T_{\mathrm{snippet}}]$. To apply the network on arbitrary length intervals $[0, T_{\mathrm{inference}}]$ we divide the interval into overlapping length T_{snippet} snippets and apply the network on each snippet individually. The snippets are combined using a Hann window ensuring smooth transitions.

Numerical experiments: After performing the training, the performance of the network is evaluated on 2D MPI system matrices, which are calculated using the Fokker-Planck solver [10] and the trained FNO. The 2D Lissajous imaging sequence (frequencies $f_x = 2.5\ \mathrm{MHz}/102$, $f_y = 2.5\ \mathrm{MHz}/96$, amplitudes $A_x, A_y = 12\ \mathrm{mT}/\mu_0$, gradient $1\ \mathrm{T/m}/\mu_0$) is chosen based on the real imaging sequence of the preclinical MPI system of Bruker [1]. We consider particles of the same 20 nm diameter as used for the test data and an anisotropy constant of $K_{\mathrm{anis}} = 1250\ \mathrm{Jm}^{-3}$. In the first setting, the easy-axis n is chosen as a gradient field, which has been shown to approximate the behavior of fluid MNP very well [1]. In the second case, we consider axis-aligned particles with a fixed easy-axis (45° and 135° w.r.t. the x-axis) in the xy plane [6]. The system matrix is sampled on a 30×30 grid including 25% overscan.

10.18416/ijmpi.2023.2303003

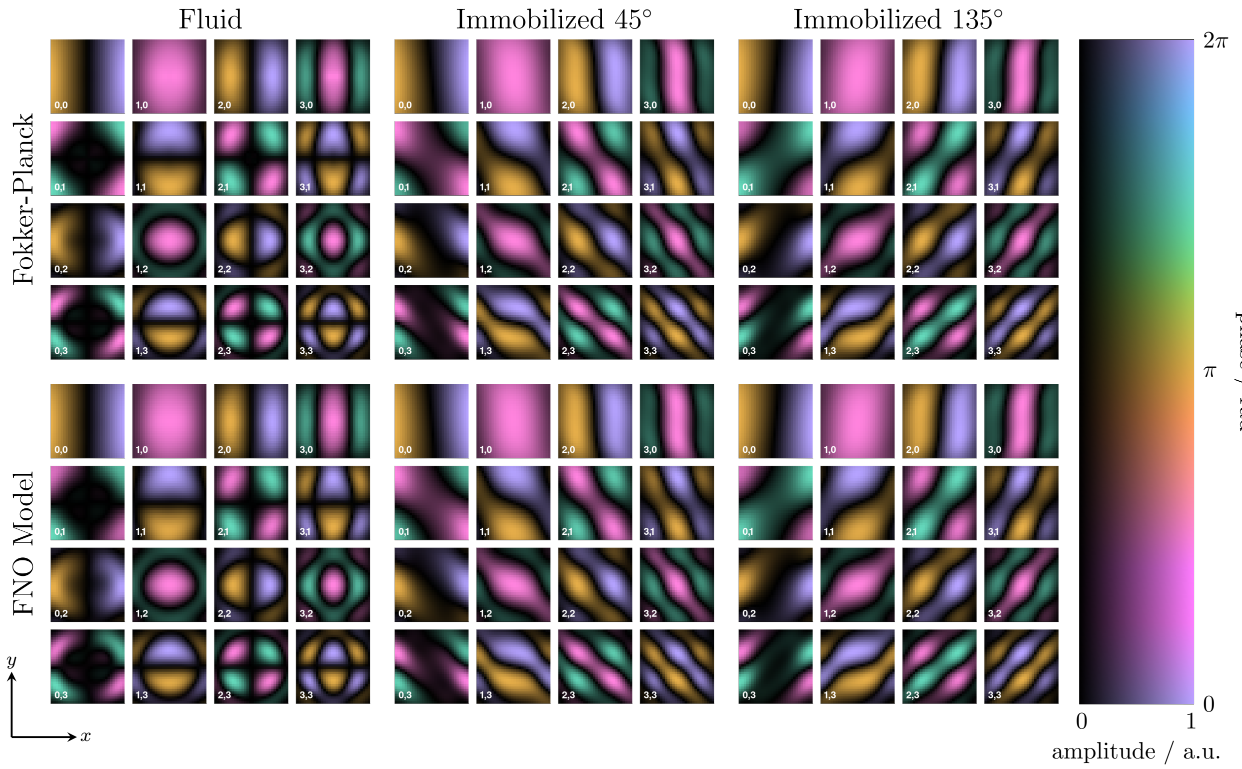

Figure 2: Simulated 2D system matrices using the physics-based Fokker-Planck model (first row) and the FNO model (second row). The first column shows simulated fluid MNPs, whereas the second and third column show axis-aligned immobilized MNPs ($\boldsymbol{n}$ rotated by 45° and 135° w.r.t. the x-axis in the xy plane). For each system matrix representative rows corresponding to mixing orders $m_x, m_x \in \{0, \dots, 3\}$ are shown. Each frequency component is normalized before applying the complex colormap shown on the right.

III. Results

Fig. 2 shows the system matrices calculated using the Fokker-Planck model and the FNO model. One can see that qualitatively, the FNO solution resembles the key characteristics of the Fokker-Planck solution very well. The mean error over all positions is 1.2 % for the fluid SM and 1.1 % (45°), 1.1 % (135°) for the immobilized SMs.

IV. Discussion and conclusion

In this work, we have shown that Fourier neural operators provide a powerful framework for accelerating the Fokker-Planck-based MNP simulation. The Fokker-Planck simulation of one 2D system matrix took 64 min on a single core of an Apple M1Max CPU. The FNO solution took just 20 s which is an acceleration of a factor 192. On a GPU (Nvidia Geforce 2080 TI), the time was further reduced to 2.7 s. For a 3D system matrix evaluated on a 30^3 grid, the simulation time went from 40 d to 5 h on the single CPU core and 25 min on the GPU. All these numbers need to be compared to 4 h (single core CPU) for generating the training data and 14 min for training on the GPU. Even when taking this time into account, it is a factor 206 faster to generate a 3D system matrix with the FNO simulator than with the Fokker-Planck Simulator.

An open question is, how the accuracy of the FNO approach is affected when training with different particle diameters and with a larger set of anisotropy constants.

Acknowledgments

H. Albers and T. Kluth acknowledge funding by the German Research Foundation (DFG, Deutsche Forschungsgemeinschaft) - project 426078691.

Author’s statement

Conflict of interest: Authors state no conflict of interest.

References

[1] 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.

[2] W. F. Brown Jr. Thermal fluctuations of a single-domain particle. *Physical review*, 130(5):1677, 1963.

[3] T. Kluth. Mathematical models for magnetic particle imaging. *Inverse Problems*, 34(8):083001, 2018.

[4] C. Shasha and K. M. Krishnan. Nonequilibrium dynamics of magnetic nanoparticles with applications in biomedicine. *Advanced Materials*, 33(23):1904131, 2021.

[5] H. Albers, T. Kluth, and T. Knopp. Simulating magnetization dynamics of large ensembles of single domain nanoparticles: Numerical study of Brown/Néel dynamics and parameter identification problems in magnetic particle imaging. *Journal of Magnetism and Magnetic Materials*, 541:168508, 2022, doi:https://doi.org/10.1016/j.jmmm.2021.168508.

[6] H. Albers, T. Knopp, M. Möddel, M. Boberg, and T. Kluth. Modeling the magnetization dynamics for large ensembles of immobilized magnetic nanoparticles in multi-dimensional magnetic particle imaging. *Journal of Magnetism and Magnetic Materials*, 543:168534, 2022, doi:https://doi.org/10.1016/j.jmmm.2021.168534.

[7] A. Anandkumar, K. Azizzadenesheli, K. Bhattacharya, N. Kovachki, Z. Li, B. Liu, and A. Stuart, Neural operator: Graph kernel network for partial differential equations, in *ICLR 2020 Workshop on Integration of Deep Neural Models and Differential Equations*, 2019.

[8] Z. Li, N. B. Kovachki, K. Azizzadenesheli, B. liu, K. Bhattacharya, A. Stuart, and A. Anandkumar, Fourier neural operator for parametric partial differential equations, in *International Conference on Learning Representations*, 2021.

[9] T. Kluth. Mathematical models for magnetic particle imaging. *Inverse Problems*, 34(8):083001, 2018.

[10] H. Albers, T. Kluth, and T. Knopp. MNPDynamics: A computational toolbox for simulating magnetic moment behavior of ensembles of nanoparticles. *International Journal on Magnetic Particle Imaging*, 6(2 Suppl 1), 2020.