IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 3 May 2024, accepted 16 May 2024, date of publication 20 May 2024, date of current version 28 May 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3403246

RESEARCH ARTICLE

Optimized Magnetic Particle Navigation Based on Data-Driven Model for Enhanced Drug Targeting

RIKKERT VAN DURME[1], GUILLAUME CREVECOEUR[1,2], (Member, IEEE), LUC DUPRÉ[1], (Member, IEEE), AND ANNELIES COENE[1,2,3]

[1]Department of Electromechanical, Systems and Metal Engineering, Ghent University, 9000 Ghent, Belgium
[2]MIRO Core Laboratory, Flanders Make, 3920 Lommel, Belgium
[3]Cancer Research Institute Ghent, 9000 Ghent, Belgium

Corresponding author: Rikkert Van Durme (rikkert.vandurme@ugent.be)

The work of Annelies Coene was supported by the Research Foundation–Flanders (FWO) through a Senior Postdoctoral Fellowship under Grant 12Z4722N.

ABSTRACT Targeted drug delivery by magnetically steering micro- and nanoparticles for specified therapy is gaining ground in the field of medical treatments. Yet fundamental challenges with regards to modeling particle movement and reaching desired regions exist. In this work, we use data-driven modeling to predict the velocities of a particle cluster from its positions and electromagnet currents in an *in-vitro* targeting setup. This explicit velocity prediction using a neural network, not found in previous works on this topic, is compared to a state-of-the-art physics-based model, and produced more accurate estimated particle cluster trajectories. Furthermore, for the first time, the data-driven model is integrated in model-based optimization methods designed to maximize the particle velocity in a predefined direction, or guide particles from an initial to a final position under minimized energy dissipation taking into account future time steps without requiring real-time feedback. Simulated and measured optimized particle trajectories strongly overlap in the experimental setup. With these findings, magnetic drug targeting can be made more accurate and brought closer to its clinical implementation.

INDEX TERMS Data-driven model, magnetic drug targeting, magnetic particle navigation, optimization.

I. INTRODUCTION

Magnetic micro- and nanoscale particles and robots are increasingly being investigated for biomedical purposes such as disease detection [1], [2] and targeted drug delivery [3], [4], [5]. The latter, a field often referred to as magnetic drug targeting (MDT), revolves around the direction of magnetic entities carrying therapeutics towards diseased regions in the human body [6]. Particles in the human body are subjected to magnetic forces to navigate through vessel networks (e.g. hepatic arteries [7]) or cross biological membranes (e.g. the blood-brain barrier [8]). The guiding or steering of these magnetic carriers is effectuated with external electromagnets to circumvent the need for invasive surgery.

The associate editor coordinating the review of this manuscript and approving it for publication was Mostafa M. Fouda.

An essential part of MDT is the prediction of the particle or microrobot movement under magnetic field gradient and bloodstream influences. Accurate models can be used in decision-making algorithms for electromagnet currents to guide particles towards desired locations or along predefined trajectories for effective targeting [9], [10], [11], [12], [13]. In literature, one identifies different force components acting upon particles and simulates the equation of motion [14], [15], [16], [17], [18], [19], [20]. These are magnetic forces, fluid forces, gravitational and buoyant force, contact forces, etc. Furthermore, diffusion and advection forces have been investigated to calculate the spatial concentration distribution of particles in the vessels as a function of time [21], [22], [23]. All models in those works are drawn from physical first principles (e.g. the magnetic dipole force model [24]) and empirical formulas (e.g. for blood viscosity [23]).

Models can also be developed from measurements. The field of machine learning (ML) is dedicated to learning behaviors and predicting values or patterns purely from data [25], [26]. Advantage of the data-driven approach is that uncertain parameter values, such as difficult to measure magnetic particle properties, do not need to be known to fit the model. In the context of modeling magnetic targeting, ML has been applied [27], [28], [29], [30]. In [27], Yu et al. used random forests and artificial neural networks (NN), two ML methods, to predict the magnetic fields in a biomedical electromagnetic navigation system with magnetic sensor arrays. The random forest and neural network outperformed the usual physics-based approach with a 40% and 87% reduction of the root mean squared error when predicting the magnetic field magnitude, respectively. Neural networks have also been employed to model the linear velocity of a swimming magnetic miniature robot, from recordings of the magnetic field, frequency, and a geometry-related angle [28]. Yang et al. predicted the behavior of swarms of particles [29]. From color camera images of current particle distributions, orientations, and environments, the particle swarm locations and orientations in the next time step are identified by a neural network trained on a dataset generated by a simulation engine. Lastly, in [30], a neural network was trained to map a relationship between coil currents and the aggregation center of microagents. These current ML implementations in MDT do not provide a clear answer to explicitly establishing a relation between the velocities of magnetic particles and applied electromagnet currents, which is useful for the prediction of trajectories and integration in particle steering algorithms. Also, collecting real-life instead of simulated data for training needs to be addressed. We develop in this work a NN estimating the velocity of a cluster of magnetic particles from a dataset of electromagnet currents, positions and velocities obtained on a self-made experimental setup. The particle velocity as explicit model output allowed a comparison with a state-of-the-art physics-based model in terms of performance and was not found in the papers on this topic. No characterization of the electromagnets and used particles was needed for the NN. Moreover, we trained the model on real-life measurements rather than a simulated environment. The accuracy of both the data-driven and the physics-based model is low on data outside the training domain. Expanding the training data range clearly improved the performance of the data-driven model, while having only a small effect on the physics-inspired model. Lastly, from data collected *in-vitro*, our model can predict the magnetic force acting on particles, which is not strongly affected by their surrounding medium. Once particles are introduced *in-vivo*, this magnetic force model remains applicable for estimating particle trajectories.

Optimization algorithms for particle navigation with the data-driven model are investigated next. The control of particle movement over time requires the actuation of optimized electromagnet currents. Under the resulting magnetic fields and gradients, the particles are able to follow a certain path or reach a point against fluid forces in a region of interest. Electromagnet activation schemes to get particles in the desired branch of bifurcated vessels have been reported to this end [31], [32], [33]. Furthermore, researchers have applied algorithms drawn from control theory. For example, Probst et al. created a 2D setup for magnetic nanoparticle steering and a feedback control method to steer particles along arbitrary trajectories [9]. Its controller design was further elaborated upon by Komaee and Shapiro [10], [11], [34]. In [12], [35], and [36], Khalil et al. showed magnetic microparticles being guided in stationary and time-varying fluid flows using proportional-derivative and optimal control. Others have implemented minimization problems to manipulate a spatial domain of multiple particles [13]. In earlier work, we have devised optimization strategies to guide particles: on the one hand the magnetic particle force along a predefined direction was maximized, and on the other hand particles in dynamically optimized magnetic fields were manipulated from an initial to a final position in a time interval [37], [38]. All of these studies make use of physics-based *in-silico* models, which represent an idealized version of reality. Real-time tracking of the particle locations is then necessary to successfully apply the steering algorithms. This is a limitation in *in-vivo* targeting problems, where obtaining accurate particle locations and distributions is a complex procedure and ongoing topic of research [39], [40]. The optimal control algorithms in [11] and [13] were not verified experimentally. In the current study, we have replaced the physical model with a data-driven model in the optimization and experimentally validate the performance on the targeting setup. Thanks to the predictive approach, no real-time position feedback is required. The data-driven model provides satisfactory reliability and as such a successful particle guiding is enabled without requiring any prior knowledge on coil geometry and magnetic particle properties, and potentially unknown parameters that occur *in-vivo*.

Other studies have applied learning-based methods for the trajectory control of (swarms of) microrobots [28], [29], [41]. In [41], a real-time servo controller is designed by training a NN on multiple desired trajectories of a microswimmer. Deep (reinforcement) learning was successfully implemented in [28] for miniature robot control and in [29] for navigating swarms of microrobots. These methods allow model-free navigation of particles under unknown flow rates and obstacles crossing their path, learned in simulated environments. The neural network trained particle/robot behavior is in all these cases immediately combined with the optimization goal and real-time tracking to get the desired magnetic fields and other inputs at each time step. We on the other hand identified a separate model for the particle movement that can be interpreted against physical models, and added a predictive aspect with multiple future time steps to control the movement of particles, also validating prior model-based control algorithms.

The structure of this paper is as follows. In Section II, we discuss the targeting setup as built in our lab, the experiments executed to obtain the data, the physics-based and neural network model, and finally the particle targeting optimization methods. Results of the neural net performance are compared to the physical model, and the optimizations are evaluated in Section III. The findings are further commented on and discussed in Section IV, with a conclusion and outlook of this work in Section V.

II. MATERIALS AND METHODS

A. SETUP AND EXPERIMENTS

In magnetic targeting, the aim is to navigate particles in a certain direction or towards a certain point. The particle trajectories result from the forces acting upon it, which are the magnetic force from field gradients, fluid drag force, particle interaction force, gravitational force, etc., and are predicted using Newton's second law of motion [18]. The magnetic field gradient is generated by nearby electromagnets carrying user-controlled currents, making it the only force component that can be adjusted directly. We write, for a cluster of aggregated particles,

$$m_p \ddot{\boldsymbol{x}} = \boldsymbol{F}_m(\boldsymbol{x}, \boldsymbol{u}) + \sum_i \boldsymbol{F}_i(\boldsymbol{x}, \dot{\boldsymbol{x}}) \tag{1}$$

where $\boldsymbol{x}$, $\dot{\boldsymbol{x}}$, $\ddot{\boldsymbol{x}}$ are respectively the position, velocity, and acceleration of the cluster center, m_p is the cluster mass, $\boldsymbol{F}_m$ the magnetic force which depends on the position and n_u controllable coil currents $\boldsymbol{u} \in \mathbb{R}^{n_u \times 1}$, and $\boldsymbol{F}_i$ expresses the remaining forces acting upon the particle cluster as a function of particle size and type, medium viscosity, etc. These relations can be modeled from physical principles to a certain level of detail, posing a trade-off between calculation time and accuracy. The most important physical models have been discussed in [19] and [20].

In this work, we consider a single particle cluster and restrict its movement to two dimensions. This is done by developing a 2D magnetic targeting setup similar to the ones found in [9], [36], and [42]. It allows particle movement in 2D with four self-wound electromagnets (200 turns, diameter 54 mm, wire thickness 1 mm, air core, resistance 1.4 Ω) surrounding a petri dish (Corning Gosselin 90 mm × 14.2 mm) containing microparticles on the water surface, as shown in Fig. 1. Each coil current is independently actuated with a bipolar DC current source (Kepco BOP36-12M) connected to a pc (Intel Core i7-8700 @ 3.20 GHz, 16 GB RAM) and dSPACE module for digital-to-analog conversion. The particles are iron oxides within a size range of 10-200 μ m. The position of the particles is recorded with a camera (Jai GO-5000C-USB with Kowa LM25HC lens, 2560 × 2048 px, 61.9 Hz) and the Python OpenCV image processing package. First, the imaging is calibrated by drawing the region of interest within the petri dish, making the center of a yellow dot the origin of the xy frame, and drawing a blue dot to define the x axis as the line between the yellow and blue dot. The actual distance between both the dots is exactly 3 cm, which provides a relation between pixel values and actual position in centimeters. The colored dot pixel positions are determined by masking the RGB color values, grayscaling, and thresholding to obtain a black blob against a white background. Then, we get the blob contour and the pixel coordinates of the blob centroid with the OpenCV functions findContours() and moments(). The pixel coordinates are then transformed into xy coordinates with respect to our frame of reference. The particle cluster xy position is acquired in the same way, by taking the centroid of the cluster's contour as coordinates. Fetching the image and computing the cluster coordinates on the connected lab pc takes 0.03 s (sampling frequency of 33 Hz and 61.9 camera frames per second). A view of all the setup parts is provided in Fig. 2.

The restriction to 2D movement allows us to validate our developed methods as a proof-of-concept with reduced complexity. Many magnetic targeting solutions have been approached in two dimensions [33], [43], [44]. A possible expansion to a setup in 3D requires at least two additional electromagnets along the z axis and one more camera for imaging a third position coordinate. Furthermore, a reservoir would be needed with a liquid medium in which particles are free to move. The same principles as described for our 2D setup hold up, albeit with more complex hardware and software due to the added z coordinate and coil currents. The novel methods presented in this research are applicable to any setup with controllable electromagnets.

For our data gathering experiments, we iteratively generate random coil currents between −4 and 4 A for 1.5 s and register the time instant and position coordinates each 0.25 s in multiple experimental runs. Our camera and pc's coordinate sampling frequency of 33 Hz surpasses the frequency of 4 Hz at which we collect data. The pc and software dependent time delay between acquiring the camera image and storing a position xy coordinate, which should be verified in each specific system and setup for this purpose, can be accounted for in the time series dataset. Most processors, cameras, and controllers can operate at high sampling rates and response times compared to the relatively low particle velocities in MDT. The particle cluster moves in one single shape as required for the continuity of the data and models, and within a radius of 18 mm from the center of the setup (the origin). This assumption holds in many *in-vivo* environments, where clusters tend to keep their shape as blood velocities are too low to break them up and conditions within the region of interest are relatively constant [45].

Particles close to one of the coils end up irrevocably trapped at the petri dish edge because of the strong attraction to the nearest coil. To avoid this, a condition was programmed to activate only the farthest coils and attract the cluster away from the domain edge back towards the center. The domain of operation is dependent on the setup and the application. A graphical representation of the experimental

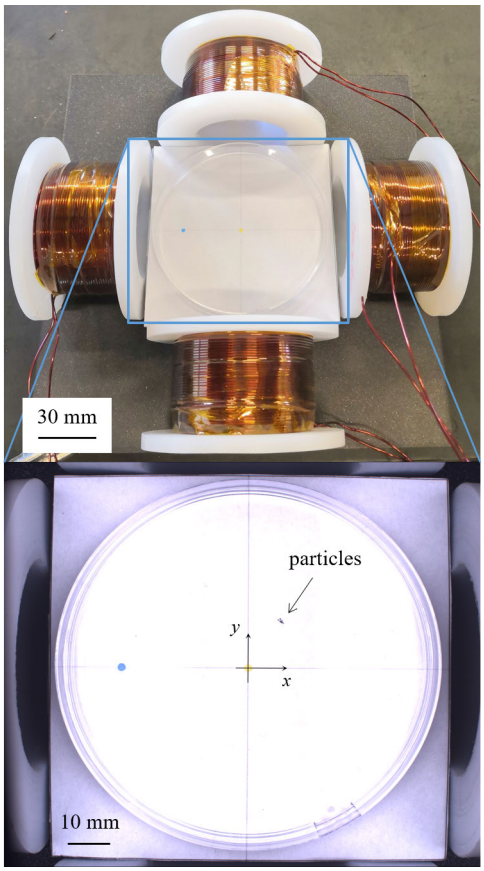

FIGURE 1. Photo of magnetic targeting setup (above) and camera image of the petri dish filled with water and particles on the surface (below). The yellow dot corresponds with the origin of the *xy* reference frame and the blue dot is on the *x* axis at 3 cm from the origin. This is used for automated camera image calibration.

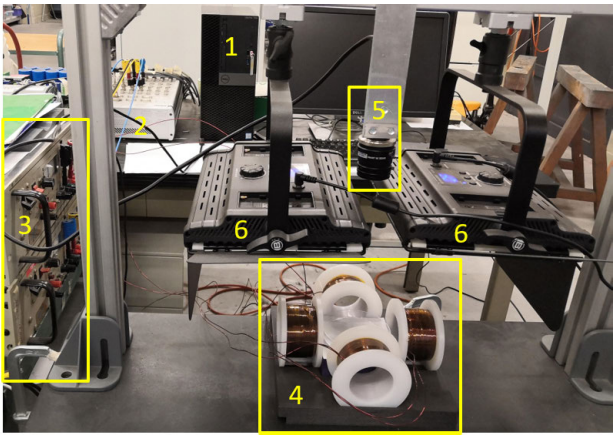

FIGURE 2. View of the full setup and its componenents. A pc (1) is connected to a dSPACE module (2). This converts digital signals to an analog signal actuating the current sources (3) and consequentially the electromagnet fields in the setup (4). Resulting magnetic gradients move the magnetic particles present in the petri dish, with positions recorded using a camera (5) connected to pc (1) for image processing. Spotlights (6) maximize the contrast and make sure that results are independent of ambient light (turned off in this photo for clarity).

data-gathering is shown in Fig. 3. The resulting dataset consists of 2D positions $\boldsymbol{x}$, four currents $\boldsymbol{u}$ ($n_u = 4$), times t, and velocities $\boldsymbol{v}$, computed through numerical differentiation $\boldsymbol{v}_k = \frac{\boldsymbol{x}_{k+1}-\boldsymbol{x}_k}{t_{k+1}-t_k}$. A total number of 15981 observations of time instants, electromagnet currents, and cluster positions was obtained over an experimental runtime of about 67 minutes. The physical validity of each observation was checked.

In the next section, we aim to predict the cluster velocity from its position and the applied electromagnet currents from the measurements.

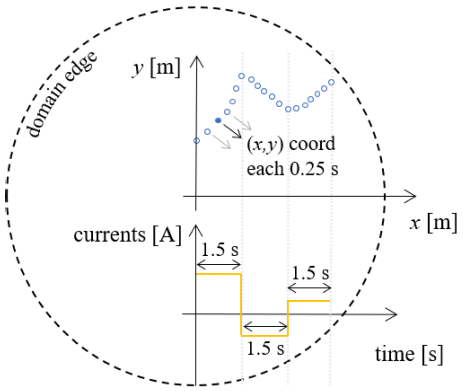

FIGURE 3. Schematic of the data-gathering experiments. Each 1.5 s, the four currents are changed to new random values, while recording the particle cluster position x each 0.25 s, corresponding to the blue circles. When the particle reaches the edge of the predefined domain, a programmed condition activates the coil on the opposite side to attract the particle back towards the center.

B. PHYSICS-BASED AND NEURAL NETWORK MODEL

A single particle or a small cluster of particles with 2D planar movement is considered in this work. The external forces acting upon the particle(s) are hence reduced to the magnetic force from field gradients [24] and the drag force $\boldsymbol{F}_d$ from the liquid medium [46]. Weight, buoyancy, interaction, and contact forces are not present or cancel each other out as the cluster moves horizontally. Thus, (1) is reduced to $m_p\ddot{\boldsymbol{x}} = \boldsymbol{F}_m + \boldsymbol{F}_d$. According to Stokes' theorem [46], the drag force on a particle in a liquid medium can be modeled as $\boldsymbol{F}_d = \zeta(\boldsymbol{v}_f - \boldsymbol{v})$, where $\boldsymbol{v} = \dot{\boldsymbol{x}}$ is the particle velocity, $\boldsymbol{v}_f$ the liquid velocity, and $\zeta = 6\pi\eta\lambda_f R_p$ depends on the liquid viscosity η, the particle radius R_p, and geometry correction factor λ_f, which equals 1 in the case of spherical particles [19], [47]. It was noticed during our data gathering experiments that the particle (cluster) quickly changes course at the instant the magnetic fields are switched. For a spherical Fe_2O_3 particle with radius $R_p = 30\ \mu$ m, density $\rho = 5.3{\cdot}10^3$ kg/m^3 in water ($\eta = 1$ mPa·s), the parameter $\zeta = 5.7{\cdot}10^{-7}$ kg/s is approximately 1000 times larger than particle mass $m_p = \rho\frac{4}{3}\pi R_p^3 = 6\cdot10^{-10}$ kg. The cluster acceleration is computed using $\boldsymbol{a}_k = \frac{\boldsymbol{v}_{k+1}-\boldsymbol{v}_k}{t_{k+1}-t_k}$ and multiplied by the mass to compare the inertia term $m_p\ddot{\boldsymbol{x}}$ to the drag force $-\zeta\dot{\boldsymbol{x}}$. The magnitude of the drag force was on average 2500 times the inertia term, and hence is negligible. The assumption of zero inertia is often made within the context of magnetic

targeting with nano- and microparticles [17], [21]. At higher viscosities, e.g. 3 mPa·s for blood, the low impact of inertia would be even more pronounced. The experiments could be conducted in the same way with a medium more viscous than water. The same methods as developed in this work would apply.

From the above analysis, we may write the particle velocity as an unknown function of the 2D particle positions and four electromagnet currents:

$$\boldsymbol{f} : \mathbb{R}^{2\times 1} \times \mathbb{R}^{4\times 1} \longrightarrow \mathbb{R}^{2\times 1}, (\boldsymbol{x}, \boldsymbol{u}) \mapsto \boldsymbol{v} = \boldsymbol{f}(\boldsymbol{x}, \boldsymbol{u})$$

$\boldsymbol{f}$ is identified on the one hand by a physics-inspired model of which the parameters are fitted to the experimental data, and on the other hand a purely data-driven regression using methods from machine learning. Approximations of $\boldsymbol{f}$ will be denoted as $\hat{\boldsymbol{v}}$, indicating that it is the estimated cluster velocity. Establishing this explicit relation is useful to assess and optimize a wide range of magnetic targeting problems, and was not found in previous papers.

The physics-based approach requires a parametrized expression for $\boldsymbol{F}_m$ next to the drag force $\boldsymbol{F}_d = -\zeta \boldsymbol{v}$. In any externally applied magnetic field $\boldsymbol{B} = \mu_0 \boldsymbol{H}$, a magnetic dipole with moment $\boldsymbol{m}$ experiences a force $\boldsymbol{F}_m = (\boldsymbol{m} \cdot \nabla)\boldsymbol{B}$ [24]. The particle cluster is approximated as a single dipole which quickly aligns with the external field $\boldsymbol{H}$ and saturates as a function of this field's magnitude. We model this through the hyperbolic tangent function with unknown parameters k_1 and k_2 to obtain $\boldsymbol{m} = k_1 \tanh(k_2 \|\boldsymbol{H}\|) \frac{\boldsymbol{H}}{\|\boldsymbol{H}\|}$ or $\boldsymbol{m} = 0$ when no field is present [48]. k_1 and k_2 need to be determined. These parameters and the relation between particle geometry, magnetization, and applied magnetic field could be characterized through precise magnetic measurements and spectral analysis [49], but here this process is avoided by our model identification approach, which is one of its great advantages.

From the force equilibrium $0 \approx \boldsymbol{F}_m + \boldsymbol{F}_d$, the particle velocity is either zero ($\boldsymbol{H} = 0$) or

$$\hat{\boldsymbol{v}}(\boldsymbol{x}, \boldsymbol{u}) = \mu_0 \zeta^{-1} k_1 \left(\tanh(k_2 \|\boldsymbol{H}\|) \frac{\boldsymbol{H}}{\|\boldsymbol{H}\|} \cdot \nabla \right) \boldsymbol{H}(\boldsymbol{x}, \boldsymbol{u}) \quad (2)$$

The field $\boldsymbol{H}$ is generated by the setup's electromagnets. Its relation with position and current path was established through Biot-Savart's law [50]. This means that, within the low frequency range, we may linearly superimpose the fields from all setup coils and write $\boldsymbol{H}(\boldsymbol{x}, \boldsymbol{u}) = \sum_{k=1}^{n_u} u_k \boldsymbol{G}_k(\boldsymbol{x}) = G(\boldsymbol{x})\boldsymbol{u}$, where $\boldsymbol{G}_k$, the k-th column of G, is dependent on position and coil geometry. We obtain semi-analytical expressions for $\boldsymbol{G}_k$ and its spatial derivative from [51], treating each of the 200 coil turns as a single loop and adding all field/gradient contributions. Rather than going through measuring with a Gaussmeter and mapping the coil field to the position, we parametrize (2) as the goal is to identify the particle velocities. Furthermore, in the case the medium viscosity is non-homogeneous in space, the relation between viscosity and position $\zeta^{-1}(\boldsymbol{x})$ should be included by the user, possibly with extra parameters to be identified.

A parametric matrix and vector are introduced

$$A = \begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \end{bmatrix} \qquad \boldsymbol{b} = \begin{bmatrix} b_1 \\ b_2 \end{bmatrix}$$

and we estimate the velocity in a data point $(\boldsymbol{x}_i, \boldsymbol{u}_i)$ by

$$\hat{\boldsymbol{v}}_i(A, k_2, \boldsymbol{b}) = A \cdot \left(\tanh(k_2 \|G(\boldsymbol{x}_i)\boldsymbol{u}_i\|) \frac{G(\boldsymbol{x}_i)\boldsymbol{u}_i}{\|G(\boldsymbol{x}_i)\boldsymbol{u}_i\|} \cdot \nabla \right) G(\boldsymbol{x}_i)\boldsymbol{u}_i + \boldsymbol{b} \quad (3)$$

Now, the task at hand is to find the best $\{A, k_2, \boldsymbol{b}\}$ such that the predicted velocity $\hat{\boldsymbol{v}}_i$ is as close as possible to the actual velocity $\boldsymbol{v}_i$. This can be done by finding the values that minimize the sum of the squares of the difference between both these velocities in each data point:

$$\{A^*, k_2^*, \boldsymbol{b}^*\} = \arg\min_{A, k_2, \boldsymbol{b}} \sum_i \|\hat{\boldsymbol{v}}_i(A, k_2, \boldsymbol{b}) - \boldsymbol{v}_i\|^2 \quad (4)$$

Since the model is expressed nonlinearly in terms of parameter k_2, this is referred to as a nonlinear least-squares fitting problem. Numerical methods to solve such optimization problems can be found in [52]. We use the MATLAB R2023b built-in function lsqcurvefit(). $\{A^*, k_2^*, \boldsymbol{b}^*\}$ is incorporated in (3) to obtain $\hat{\boldsymbol{v}}(\boldsymbol{x}, \boldsymbol{u})$. If our proposed physical model (2) is accurate, A^* would be a multiple of the identity matrix and $\boldsymbol{b}^*$ would be zero. We do expect some mixing of terms and asymmetries, which result in non-zero off-diagonal elements in A^* and non-zero elements in $\boldsymbol{b}^*$.

Alternatively, a machine learning regression is done on the experimental data. This is a black-box approach in which the parameters of a chosen general ML model, instead of a specific physical model, are optimized for. Examples of such models are random forests, support vector machines, gradient boosted regression trees, neural networks, etc. [53]. In this work, we opted for a neural network because of its straightforward implementation [54], [55]. Now, instead of (2), we have

$$\hat{\boldsymbol{v}}(\boldsymbol{x}, \boldsymbol{u}) = \boldsymbol{N}(\boldsymbol{x}, \boldsymbol{u}) \quad (5)$$

where $\boldsymbol{N}$ represents the trained NN. To train it, we performed a 5-fold cross-validation on the experimental observations, leaving out 20% for testing, and splitting the remaining data each time in 85% training and 15% validation sets. A range of fully-connected hidden layers (1, 2, 3) and neurons (4 to 128 per layer), activation functions (ReLU, hyperbolic tangent, and sigmoid), and numbers of epochs (max. 1000) was swept through for hyperparameter tuning. The network weight and bias optimizations were executed with the Levenberg-Marquardt algorithm in MATLAB. We also compared between normalized and original input data. Lastly, we trained networks without and with simulated artificial noise added to the input and output over a range of 10 to 40 dB signal-to-noise ratios, because there is a possibility of reducing overfitting [56]. The performance criterion was

$R^2 = 1 - \sum_i \|\boldsymbol{v}_i - \hat{\boldsymbol{v}}_i\|^2 / \sum_i \|\boldsymbol{v}_i - \bar{\boldsymbol{v}}\|^2$ with $\bar{\boldsymbol{v}}$ the mean of the observed velocities.

A comparison with random forests, embedded trees, gradient booster machines (hyperparameter tuning with respect to bag or LSBoost, minimum leaf size, number of learners, learning rate), and support vector machines (optimized for different kernel functions) was made afterwards in the Regression Learner Toolbox in MATLAB R2023b. The meaning of these models and hyperparameters is not discussed here, but can be found in machine learning handbooks [57].

The setup with coils, currents, particle positions, physics-based model, and NN is visualized in Fig. 4. The NN's weights and biases amount to much more parameters than the physics-based model and as such its training has a higher computational cost. Because of the inherent nonlinearity of NNs and their activation functions, we will be able to capture unexpected dynamics of the particle movement as a function of the currents without characterizing the electromagnets and magnetic particles.

Predictive quality of the physics-identified model in (3) and the neural network are compared. It is generally known that purely data-driven models perform poorer outside the training data domain. In other words, they hold up well for interpolation but less so for extrapolation. Physics-based models on the contrary do not suffer from this drawback so long as the captured physical laws remain valid. We compare the behavior for inter- and extrapolation. Firstly, we restrict the position data to within a radius of 12 mm from the domain center, identify the models based on these data, and evaluate performance on unseen test trajectories inside and outside this radius. Secondly, we expand the data seen by the models to reach 18 mm and compare performance with the first case.

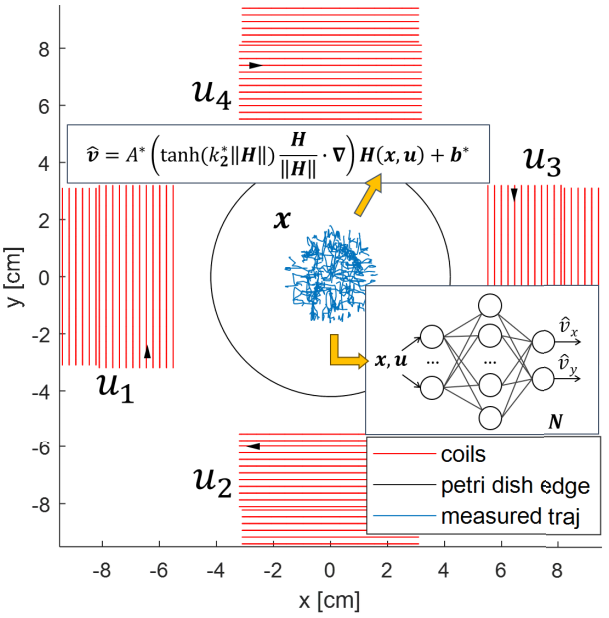

FIGURE 4. Drawing of the lab setup as in Fig. 1 with part of the measured trajectories for training or least-squares optimization (in blue). The four currents $u = (u_1, u_2, u_3, u_4)$ and particle xy coordinates x are fed to the neural network N or the parametrized physical model (3) to predict the particle velocity $\hat{v}$.

C. OPTIMIZATION OF PARTICLE MOVEMENT

1) FORCE MAXIMIZATION ALONG DEFINED DIRECTION

In our previous paper, we investigated ways to optimize the movement of particles through the bloodstream [38]. Challenges are being able to direct the particles optimally towards the desired vessel branches and reducing particle swarm scattering effects. We formulated and implemented a constrained optimization of the magnetic force on a particle along a predefined direction $\boldsymbol{n}$ with respect to the electromagnet currents. In this work, we employ the estimated velocity model $\hat{\boldsymbol{v}}$. The aim is to find the coil currents $\boldsymbol{u}^*$ that maximize the particle velocity magnitude at a known position $\boldsymbol{x}_0$, enforced along a predefined target direction through the equality constraint $\boldsymbol{v}(\boldsymbol{u})\cdot\boldsymbol{n} = \|\boldsymbol{v}(\boldsymbol{u})\|\|\boldsymbol{n}\|$. This target direction is a set by a user based on the desired particle movement. Furthermore, a boundary on the velocity divergence was added. Divergence of a vector field in a certain point is a measure for the extent to which neighbouring vectors are pointed away from each other. As such, the force vectors are more parallel and spreading of the particles within the region is limited, increasing their numbers in the desired directions. Put together,

$$\begin{aligned} &\text{find} \quad \boldsymbol{u}^* \\ &= \arg\max_{\boldsymbol{u}} \|\hat{\boldsymbol{v}}(\boldsymbol{x}_0, \boldsymbol{u})\| \\ &\quad \text{subject to} \ \ \hat{\boldsymbol{v}}(\boldsymbol{x}_0, \boldsymbol{u})\cdot\boldsymbol{n} = \|\hat{\boldsymbol{v}}(\boldsymbol{x}_0, \boldsymbol{u})\|\|\boldsymbol{n}\| \ \text{target direction} \\ &\qquad\qquad |u_k| \le u_{\max},\ k = 1, \ldots, 4 \ \text{max. current} \\ &\qquad\qquad \nabla\cdot\hat{\boldsymbol{v}}(\boldsymbol{x}_0, \boldsymbol{u}) \le d_{\max} \ \text{max. divergence} \end{aligned} \tag{6}$$

where $u_{\max}$ is the limit of the electromagnet current and $d_{\max}$ the limit to the divergence. The optimization problem is solved with MATLAB and validated on our experimental setup for different target directions. From validation on this or other lab setups, the method can be expanded to more realistic targeting situations. It justifies the combination of regression models with optimization to guide particles more efficiently through the vascular network than non-optimized alternatives.

2) PARTICLE TRAJECTORY DYNAMIC OPTIMIZATION

It has been researched how magnetic particles and particle collections can be manipulated towards a target region within a certain time interval using time-varying magnetic fields and optimal control [12], [37], [58]. We have formulated an optimization problem to guide particles from an initial to a final position $\boldsymbol{x}_0$ and $\boldsymbol{x}_f$ as defined by the user based on the targeting requirements. The current depending on the time t, $\boldsymbol{u}(t)$, is the target variable. During the time interval $[0, t_f]$, t_f being the final time, the movement of the particle

at location $\boldsymbol{x}(t)$ is governed by the modeled velocity as a function of the location and currents, $\hat{\boldsymbol{v}}(\boldsymbol{x}(t), \boldsymbol{u}(t))$. A cost function related to the problem can then be minimized. In our case, we minimize the energy dissipated in the coils, which is directly proportional to the sum of the squared coil currents. The dynamic optimization is run - 'dynamic' referring to the time dependency of the optimized currents - as follows,

$$\begin{aligned} &\text{find} \quad \boldsymbol{u}^*(t) \\ &= \arg\min_{\boldsymbol{u}(t)} \int_0^{t_f} \boldsymbol{u}(t)^T \boldsymbol{u}(t) dt \\ &\text{subject to} \\ &\qquad \boldsymbol{x}(0) = \boldsymbol{x}_0 \text{ initial position} \\ &\qquad \boldsymbol{x}(t_f) = \boldsymbol{x}_f \text{ target position} \\ &\qquad \dot{\boldsymbol{x}}(t) = \hat{\boldsymbol{v}}(\boldsymbol{x}(t), \boldsymbol{u}(t)) \; t \in [0, t_f] \\ &\qquad |u_k(t)| \leq u_{\max}, \; k = 1, \ldots, 4 \; t \in [0, t_f] \end{aligned} \tag{7}$$

This problem can be solved with the method of Direct Multiple Shooting (DMS) [59]. First, the current signals are split up in N_i piecewise constant pieces within $[0, t_f]$, $\boldsymbol{u}(t) = \boldsymbol{q}_i$ for $t \in [t_i, t_{i+1}]$, $i = 0, \ldots, N_i - 1$. Then, the following ordinary differential equation is solved on each interval, with artificial initial value $\boldsymbol{s}_i$,

$$\begin{aligned} &\dot{\boldsymbol{x}}_i(t) = \hat{\boldsymbol{v}}(\boldsymbol{x}_i(t), \boldsymbol{q}_i), \qquad t \in [t_i, t_{i+1}] \\ &\boldsymbol{x}_i(t_i) = \boldsymbol{s}_i \end{aligned}$$

Meanwhile, the integrals related to the cost function are computed

$$l_i = \int_{t_i}^{t_{i+1}} \boldsymbol{q}_i^T \boldsymbol{q}_i \, dt = \boldsymbol{q}_i^T \boldsymbol{q}_i (t_{i+1} - t_i)$$

and we impose continuity of the state by setting $\boldsymbol{s}_{i+1} = \boldsymbol{x}_i(t_{i+1})$ because the cluster movement is by nature continuous in time. Now, we can reformulate the problem as a constrained optimization, with $\boldsymbol{s} = [\boldsymbol{s}_0^T \cdots \boldsymbol{s}_{N_i}^T]^T$ and $\boldsymbol{q} = [\boldsymbol{q}_0^T \cdots \boldsymbol{q}_{N_i-1}^T]^T$,

$$\begin{aligned} &\text{find} \\ &\min_{\boldsymbol{s},\boldsymbol{q}} \sum_{i=0}^{N_i-1} l_i \\ &\text{subject to} \\ &\qquad \boldsymbol{s}_0 = \boldsymbol{x}_0 \text{ initial value} \\ &\qquad \boldsymbol{s}_{N_i} = \boldsymbol{x}_f \text{ terminal value} \\ &\qquad \boldsymbol{s}_{i+1} = \boldsymbol{x}_i(t_{i+1}) \; i = 0, \ldots, \; N_i - 1 \text{continuity} \\ &\qquad |q_{i,k}(t)| \leq u_{\max} \; i = 0, \ldots, N_i - 1 \text{ max. current} \\ &\qquad\qquad\qquad k = 1, \ldots, 4 \end{aligned} \tag{8}$$

which can be solved numerically using the constrained nonlinear optimization method of sequential quadratic programming [60]. The total number of target variables for this optimization problem with $n_u = 4$ current inputs and $n_x = 2$ position coordinates in $N_i = 4$ time steps is $(N_i + 1)n_x + N_i n_u = 26$.

We evaluate the performance of the algorithm on the targeting setup. The advantage of this method is that it can minimize a cost over a time interval and takes into account multiple future time steps and not just a single step based on real-time position measurements that may be hard to come by in *in-vivo* situations. Moreover, the algorithm is adjustable with different constraints and cost functions, e.g. we could minimize the magnetic field norm $\|\boldsymbol{H}(\boldsymbol{u})\|^2$, prevent trajectories from passing through certain regions, attach a cost to reaching the target, etc.

III. RESULTS

A. PHYSICS-BASED AND NN MODEL PERFORMANCE

The optimization (4) is executed on the dataset and the optimal parameters $\{A^*, k_2^*, \boldsymbol{b}^*\}$ were found to be $k_2^* = 0.0754$,

$$A^* = \begin{bmatrix} 1.45 & -0.0429 \\ -0.0362 & 1.48 \end{bmatrix} 10^{-6}, \quad \boldsymbol{b}^* = -\begin{bmatrix} 1.93 \\ 0.251 \end{bmatrix} 10^{-5}$$

A^* shows similarity with the identity matrix and $\boldsymbol{b}^*$ is orders of magnitude smaller than the measured velocities (order of mm/s vs. μ m/s), such that the proposed model (2) has a bearing on reality. The training data $R^2 = 0.33$ was actually outperformed by the testing $R^2 = 0.36$ but these values are too low for reliable predictions. The number of data included in the optimization had little effect on the parameter outcome. On the data-driven side, the best performing network $\boldsymbol{N}$ had 2 hidden ReLU layers with 32 neurons and batch normalization at the input layer. Adding noise to input and output data for training did not make the model generalize better. On average R^2 was around 0.92 on the training data and 0.90 on the test data, indicating a good generalization. With the Regression Learner toolbox we found that ensemble trees and support vector machines were the best alternative ML models. Hyperparameter optimization of ensemble trees with respect to bag or LSBoost, minimum leaf size, number of learners, and learning rate, yielded $R^2 = 0.86$ (0.85 on test data) and support vector machines with Gaussian kernel function yielded $R^2 = 0.84$ on training and test data. The performance of the neural net and the physics-based model predicting the particle velocity $\hat{\boldsymbol{v}}$ is validated by taking a test trajectory and its current sequence $\boldsymbol{u}$ unseen by the model, applying the same current for each step k from the initial position $\boldsymbol{x}_0$, and executing

$$\boldsymbol{x}_{k+1} = \boldsymbol{x}_k + (t_{k+1} - t_k) \, \hat{\boldsymbol{v}}(\boldsymbol{x}_k, \boldsymbol{u}_k), \quad k = 0, \ldots, 16. \tag{9}$$

The 16 time steps correspond with a time span of 4 s, allowing the error to propagate. It is unavoidable that the predicted trajectories diverge from the actual ones over time. Fig. 5 shows predicted unseen trajectories resulting from this iteration scheme. On the left the training data were restricted to 12 mm, on the right to 18 mm. The overlap between actual and neural net trajectories is quite strong within the training

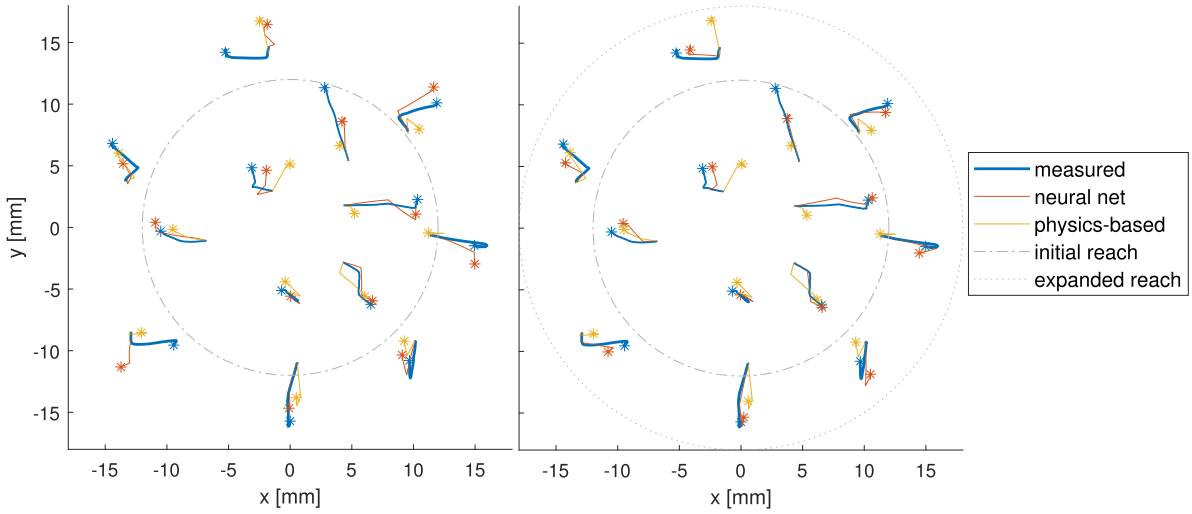

FIGURE 5. Sequences of measured particle trajectories (blue) with their neural net (red) and physics-based (yellow) predicted trajectories over 4 s, the final position indicated by a star '*'. Overall the neural net performs better than the physics-based model. On the left, the trajectories outside the training domain are more likely to be wrong for both neural net and physics-based model, whereas on the right, where the training domain is expanded, the neural net improved and outperforms the physics-based model most of the time.

domain, whereas some of them are off when extrapolated outside this domain. As expected, including more data makes improvements, seen on the right. The physics-based model however has very poor overlap on some of the trajectories both inside and outside the training domain. This is implied by the low R^2 values. On a small number of trajectories, the physics-based model is slightly better. Overall, the best practice would be to use the purely data-driven method with a representative data reach.

In the following, we apply the neural net to the optimization problems explained in Section II-C while staying inside the training domain. Optimization based on the physics-based model would necessarily generate poorer results because of its inferior predictive accuracy.

B. OPTIMIZATION RESULTS

1) VELOCITY MAXIMIZATION

We use MATLAB built-in gradient-based optimization to solve (6) with

$$\boldsymbol{x}_0 = [0\text{ mm}; 0\text{ mm}],\ d_{\max} = 10\text{ s}^{-1},$$
$$\hat{\boldsymbol{v}}(\boldsymbol{x}_0, \boldsymbol{u}) = \boldsymbol{N}(\boldsymbol{x}_0, \boldsymbol{u}),\ u_{\max} = 4\text{ A}.$$

The divergence limit $d_{\max}$ was determined through trial-and-error. We execute the optimization for the initial position $\boldsymbol{x}_0$ three times for three different target directions $\boldsymbol{n}$. The particle cluster is subjected to the optimized electromagnet currents near the origin and its positions are registered over a duration of 2 s. The trajectories are depicted in Fig. 6. Afterwards, we calculated and plotted the neural net velocity vector for one position in each measured trajectory and included the angle between the target direction and the cluster velocity direction. It can be seen that the particle cluster in different positions moves in the desired direction.

2) DYNAMIC OPTIMIZATION TOWARDS TARGET POSITION

The dynamic optimization (7) is validated using DMS with an in-house developed toolbox for MATLAB, for

$$\boldsymbol{x}_0 = [0\text{ mm}; 0\text{ mm}],\ \boldsymbol{x}_f = [5\text{ mm}; 5\text{ mm}],$$
$$\hat{\boldsymbol{v}}(\boldsymbol{x}, \boldsymbol{u}) = \boldsymbol{N}(\boldsymbol{x}, \boldsymbol{u}),\ t_f = 6\text{ s},\quad u_{\max} = 4\text{ A}.$$

The experimental results shown here are for discretization of the time in four steps, meaning that the coil currents remain constant in time intervals of 1.5 s. There is a trade-off between increased computational time and improved optimality of the result with increased number of considered time steps [59]. The relatively slow particle velocities (order of mm/s) do not require rapid altering of the currents and magnetic fields, and as such the rather low number of four time steps already provided good experimental results.

We apply the obtained current signal $\boldsymbol{u}^*(t)$ to the setup coils and demonstrate the path of the particle cluster with camera snapshots in Fig. 7. The cluster successfully ends up at the imposed target position at [5 mm; 5 mm]. For further validation, we chose two additional target positions $\boldsymbol{x}_f$ to be reached: [−9 mm; −5 mm] with $t_f = 8$ s, and [−2 mm; −8 mm] with $t_f = 7$ s. In Fig. 8, we repeated experiments with the cluster starting in the origin subjected to the optimized currents on the left. The trajectories in the center column were measured in separate runs. In the right column, we show the distance to the target position over time and the mean and standard deviation of the three repeated measured sequences. Most of the measured paths lead towards the target position along the optimized trajectory, with some deviations due to the slight discrepancies in initial cluster positions and

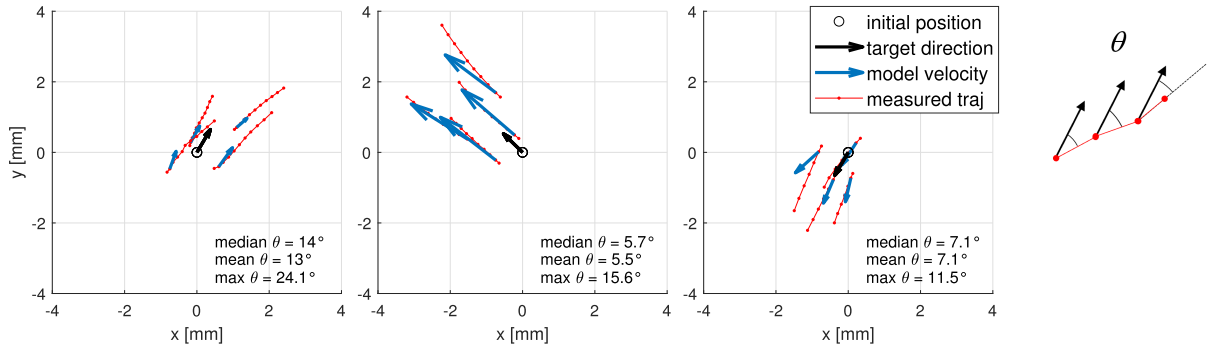

FIGURE 6. Numerical optimization of the particle velocity along three different target directions (black arrow), each computed for position [0; 0]. Four actual starting positions of the particle cluster are near the origin in each experiment. Recorded trajectories over a time span of 2 s are shown in red. The blue arrows depict the velocity as predicted by the neural net in the arrows' starting points. The positive angle between the target direction and the velocity in each recorded position, shown graphically on the right, is denoted θ. Its mean values in degrees provide a measure for the deviation of the actual trajectory from the target direction.

orientations between experiments. The difference in initial magnetic force then causes the clusters to be on different trajectories. This effect is the most pronounced in Fig. 8(c), where coil 2 has almost the maximum current and thus a large attractive force during the full interval, leading to the particle cluster going too far. The general trend is however close to the simulated path.

Movies of these experiments are supplementary material to this article.

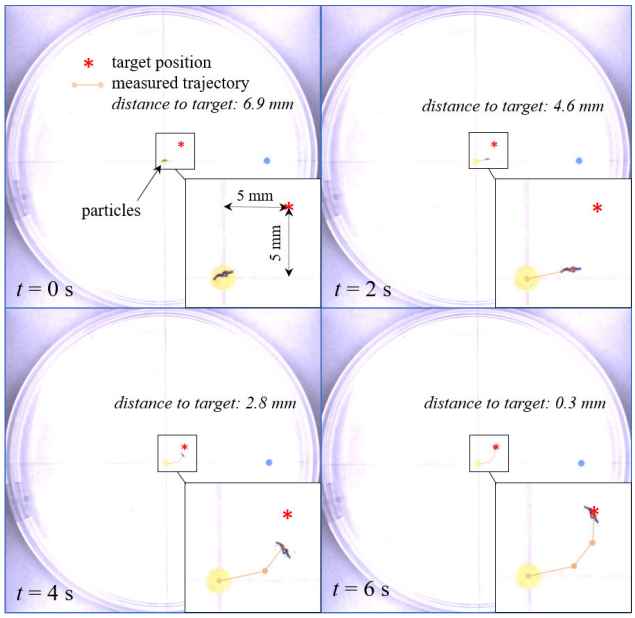

FIGURE 7. Snapshots of the particle cluster moving from the origin towards the set target [5 mm; 5 mm] under the magnetic currents obtained from dynamic optimization in 6 s. The trained neural network was used in the optimizer to predict the future particle positions.

IV. DISCUSSION

Fig. 5 demonstrated that there is a strong overlap between the measured and the NN-based trajectories, especially considering the propagation error when applying iteration scheme (9). The same level of accuracy was not achieved with the physics-based model, from which we infer that the proposed parametrized model in (3) does not sufficiently capture the behavior due to setup and particle-related uncertainties. The advantage of the black-box approach is that we do not need to clear up these uncertainties in order to have a good predictive quality.

Compared to the tuning time of the neural net, the identification of the parameters in the physical model was computationally faster (10 min. vs. a couple of hours for in-depth hyperparameter tuning) due to its smaller number of parameters than the number of NN weights and biases. On our machine with MATLAB R2023b, evaluating the NN in a single data point, i.e. generating the output velocity from the 4 input currents and 2 position coordinates, took 10 ms on average, whereas the physics-based model output was computed in only 1 ms. It is noted that these response times are heavily affected by the hardware and software and the user-defined level of detail (number of neurons, depth of the physics, etc.).

The model is trained on data from a lab setup with stationary liquid. This can be extended with phantom vessels and a pumping system for moving liquids to mimick the vascular network. In the end, particles are to be employed in the human body. We can use setup data to identify the magnetic force on the cluster, which remains unaffected by non-magnetic biological environments. According to the force equilibrium in our setup, we may assume that the velocity scales linearly with the magnetic force $\boldsymbol{v}(\boldsymbol{x},\boldsymbol{u}) = \zeta^{-1}\boldsymbol{F}_m(\boldsymbol{x},\boldsymbol{u}) = \boldsymbol{N}(\boldsymbol{x},\boldsymbol{u})$. With an accurate estimate of $\zeta = 6\pi\eta\lambda_f R_p$, we obtain a model for the magnetic force $\boldsymbol{F}_m(\boldsymbol{x},\boldsymbol{u}) = \zeta\,\boldsymbol{N}(\boldsymbol{x},\boldsymbol{u})$ specific to the coil configuration and particle size/type but not to its environment. This can then be used in combination with flow models to predict and control particle movement. Furthermore, the magnetic force sense and direction does not depend on the cluster size, only the magnitude does. We could thus multiply the magnetic force model with an empirically determined factor for different cluster sizes.

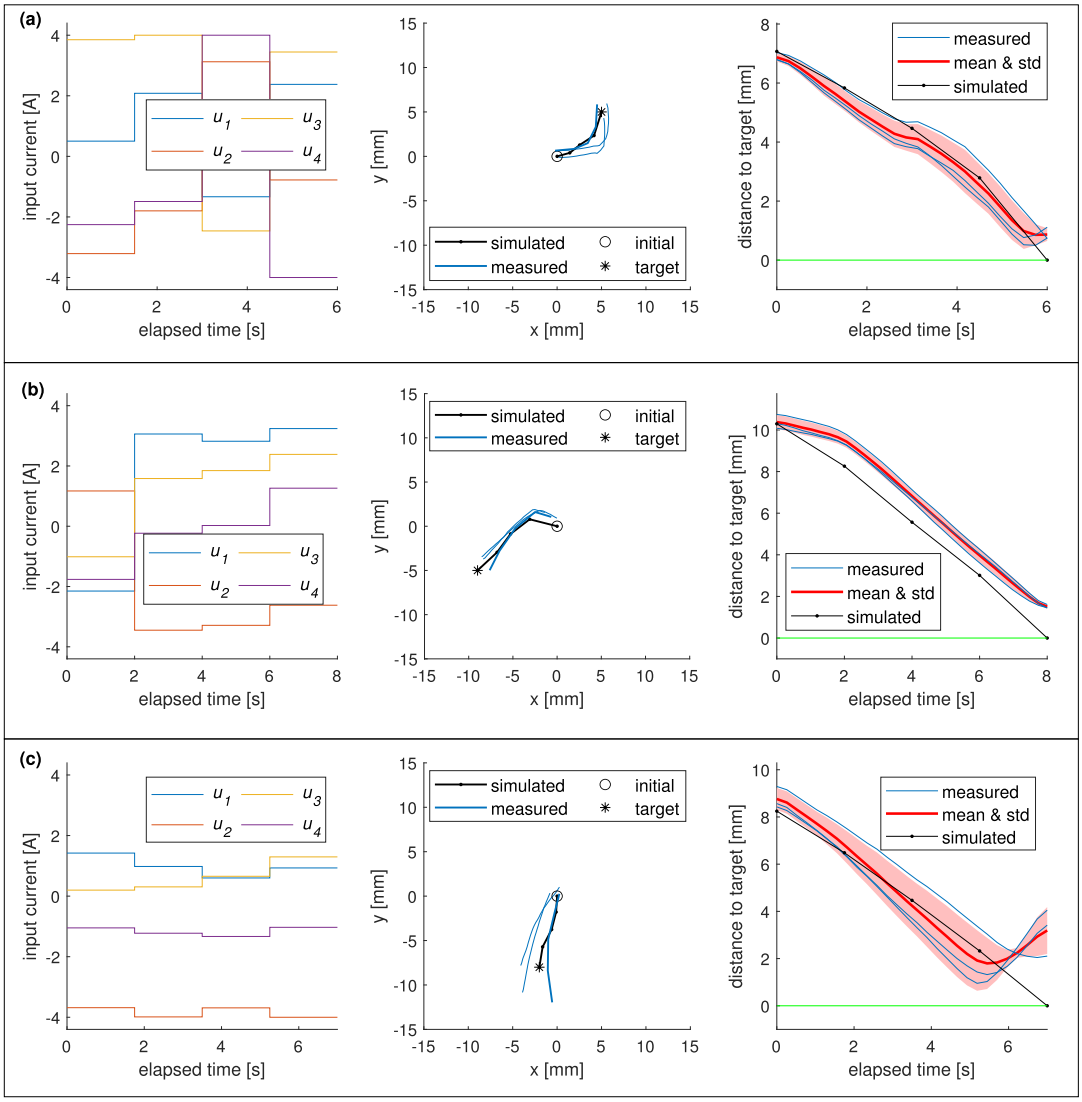

FIGURE 8. Optimized currents corresponding with the coils in Fig. 4 for three different target positions x_f and final times t_f on the left. Simulated trajectories from the DMS algorithm are included in the center in black, and measured cluster trajectories from separately repeated experiments in blue. On the right, the distance to the target for each of these three trajectories is shown, including the averaged value and standard deviation. The deviation from the target in experiment (c) can be explained by a slight inconsistency in the velocity estimation when the current in coil 2 is at its maximum value of 4 A.

Both the velocity maximization in a fixed point and the dynamic optimization over a time period with DMS show particle progression towards the set target in simulated and real-life *in-vitro* experiments. The measured particle paths in Fig. 6 move tangential to the modeled velocity arrows almost in every point. In some points, slight differences between path and modeled velocity can be noticed as a consequence of imperfect predictions. It is as expected that the farther away from the point in which the velocity was maximized, i.e. the origin, the more the trajectory deviates from the target direction. Overall, the particle movement is in the desired direction (small θ) at optimal velocity magnitude. In new particle positions, the magnetic force could be maximized with respect to updated target directions. The method is non-precise, but can be a fast and useful tool when an influx of particles is to be guided towards a desired vessel branch as discussed in [38]. Maximizing the velocity in a specific direction can be applied to specific parts of the vascular network to increase the local presence of particles. The DMS algorithm provides the required currents to guide particles towards target locations without real-time position feedback, as proven in Figs. 7-8. Even though no actual measurements are incorporated in the control during the movement, there is a significant overlap between simulated and measured trajectories in the optimizations in Fig. 8, where deviations were mostly due to differences in initial cluster position and orientation. We again note that the neural net performs well on unseen data, as the measured optimized positions

and currents were not part of the training set. Using a data-driven model instead of physics-based models as in [9], [10], and [12] is thus advised.

The time-dependent predictive optimization of cluster trajectories paves the way to more accurate steering in biological environments when real-time feedback is unavailable. Camera feedback is not possible if particles are covered by human body parts. To tackle this, advanced magnetic imaging techniques have been developed and are an ongoing topic of research. Examples are magnetic resonance and magnetic particle imaging [1], [61], of which the combination with navigating particles has been researched [7], [43], [62], [63], [64]. A MPI sequence could be succeeded by an open-loop targeting sequence with the currents computed by our methods, outperforming the dynamic optimization methods described in [11], [12], and [13]. Subsequent imaging can inform control input updates at later time instants to further guide the particles.

Other works have successfully applied NNs and deep reinforcement learning methods to navigate magnetic microrobots or particle swarms along trajectories inside blood vessels and past obstacles [28], [29], [41]. These procedures rely on real-time camera images and do not take into account multiple future time steps for their control action.

In future work, several additions and extensions for *in-vitro* validation are required before making the step towards *in-vivo* targeting. We need to apply the regression models to realistic settings in 2D and 3D, taking into account flow rates and wall effects. Advanced learning algorithms can be incorporated to transfer a prior model to more realistic setups and conditions [65]. Finally, we should make the control algorithm compatible with particle imaging techniques to enhance the goal-reaching over longer distances (for example through model-based predictive control [66]).

V. CONCLUSION

Magnetic microparticles are steered from a distance by magnetic field gradients from electromagnets in an *in-vitro* targeting setup. We presented how the velocity of a particle cluster can be predicted from its position and electromagnet currents using a data-driven neural network, and compared its performance to a physics-based model. The actually measured particle trajectories showed a strong overlap with the trajectories predicted by the neural network, significantly more so than with the physical model.

Model-based numerical optimization of the particle velocity and trajectories from initial to final position in a time interval with minimized energy dissipation was executed. The neural network served as model. Resulting simulated velocities and particle trajectories demonstrated a strong agreement with the actual recorded particle paths and velocity in the targeting setup.

The use of data-driven methods combined with optimization algorithms can advance magnetic particle-based targeted therapy towards its clinical application. In subsequent steps, the approach reported in this work is to be applied to a variety of *in-vitro* and *in-vivo* setups and imaging modalities.

REFERENCES

[1] E. Y. Yu, M. Bishop, B. Zheng, R. M. Ferguson, A. P. Khandhar, S. J. Kemp, K. M. Krishnan, P. W. Goodwill, and S. M. Conolly, "Magnetic particle imaging: A novel in vivo imaging platform for cancer detection," *Nano Lett.*, vol. 17, no. 3, pp. 1648–1654, Mar. 2017.

[2] L. C. Wu, Y. Zhang, G. Steinberg, H. Qu, S. Huang, M. Cheng, T. Bliss, F. Du, J. Rao, G. Song, L. Pisani, T. Doyle, S. Conolly, K. Krishnan, G. Grant, and M. Wintermark, "A review of magnetic particle imaging and perspectives on neuroimaging," *Amer. J. Neuroradiology*, vol. 40, no. 2, pp. 206–212, Feb. 2019.

[3] B. J. Nelson, I. K. Kaliakatsos, and J. J. Abbott, "Microrobots for minimally invasive medicine," *Annu. Rev. Biomed. Eng.*, vol. 12, no. 1, pp. 55–85, Jul. 2010.

[4] T. Vangijzegem, D. Stanicki, and S. Laurent, "Magnetic iron oxide nanoparticles for drug delivery: Applications and characteristics," *Expert Opinion Drug Del.*, vol. 16, no. 1, pp. 69–78, Jan. 2019.

[5] A. Coene and J. Leliaert, "Magnetic nanoparticles in theranostic applications," *J. Appl. Phys.*, vol. 131, no. 16, Apr. 2022, Art. no. 160902.

[6] B. Shapiro, S. Kulkarni, A. Nacev, S. Muro, P. Y. Stepanov, and I. N. Weinberg, "Open challenges in magnetic drug targeting," *WIREs Nanomedicine Nanobiotechnology*, vol. 7, no. 3, pp. 446–457, May 2015.

[7] N. Li, C. Tous, I. P. Dimov, D. Cadoret, P. Fei, Y. Majedi, S. Lessard, Z. Nosrati, K. Saatchi, U. O. Häfeli, A. Tang, S. Kadoury, S. Martel, and G. Soulez, "Quantification and 3D localization of magnetically navigated superparamagnetic particles using MRI in phantom and swine chemoembolization models," *IEEE Trans. Biomed. Eng.*, vol. 69, no. 8, pp. 2616–2627, Aug. 2022.

[8] S. D. Kong, J. Lee, S. Ramachandran, B. P. Eliceiri, V. I. Shubayev, R. Lal, and S. Jin, "Magnetic targeting of nanoparticles across the intact blood–brain barrier," *J. Controlled Release*, vol. 164, no. 1, pp. 49–57, Nov. 2012.

[9] R. Probst, J. Lin, A. Komaee, A. Nacev, Z. Cummins, and B. Shapiro, "Planar steering of a single ferrofluid drop by optimal minimum power dynamic feedback control of four electromagnets at a distance," *J. Magn. Magn. Mater.*, vol. 323, no. 7, pp. 885–896, Apr. 2011.

[10] A. Komaee and B. Shapiro, "Steering a ferromagnetic particle by optimal magnetic feedback control," *IEEE Trans. Control Syst. Technol.*, vol. 20, no. 4, pp. 1011–1024, Jul. 2012.

[11] A. Komaee, "Feedback control for transportation of magnetic fluids with minimal dispersion: A first step toward targeted magnetic drug delivery," *IEEE Trans. Control Syst. Technol.*, vol. 25, no. 1, pp. 129–144, Jan. 2017.

[12] I. S. M. Khalil, H. Abass, M. Shoukry, A. Klingner, R. M. El-Nashar, M. Serry, and S. Misra, "Robust and optimal control of magnetic microparticles inside fluidic channels with time-varying flow rates," *Int. J. Adv. Robotic Syst.*, vol. 13, no. 3, p. 123, May 2016.

[13] H. Antil, R. H. Nochetto, and P. Venegas, "Controlling the Kelvin force: Basic strategies and applications to magnetic drug targeting," *Optim. Eng.*, vol. 19, no. 3, pp. 559–589, Sep. 2018.

[14] A. D. Grief and G. Richardson, "Mathematical modelling of magnetically targeted drug delivery," *J. Magn. Magn. Mater.*, vol. 293, no. 1, pp. 455–463, May 2005.

[15] E. P. Furlani and Y. Sahoo, "Analytical model for the magnetic field and force in a magnetophoretic microsystem," *J. Phys. D, Appl. Phys.*, vol. 39, no. 9, pp. 1724–1732, May 2006.

[16] E. J. Furlani and E. P. Furlani, "A model for predicting magnetic targeting of multifunctional particles in the microvasculature," *J. Magn. Magn. Mater.*, vol. 312, no. 1, pp. 187–193, May 2007.

[17] E. P. Furlani and K. C. Ng, "Analytical model of magnetic nanoparticle transport and capture in the microvasculature," *Phys. Rev. E, Stat. Phys. Plasmas Fluids Relat. Interdiscip. Top.*, vol. 73, no. 6, Jun. 2006, Art. no. 061919.

[18] E. M. Cherry, P. G. Maxim, and J. K. Eaton, "Particle size, magnetic field, and blood velocity effects on particle retention in magnetic drug targeting," *Med. Phys.*, vol. 37, no. 1, pp. 175–182, Jan. 2010.

[19] P. Vartholomeos and C. Mavroidis, "In silico studies of magnetic microparticle aggregations in fluid environments for MRI-guided drug delivery," *IEEE Trans. Biomed. Eng.*, vol. 59, no. 11, pp. 3028–3038, Nov. 2012.

[20] H. Bayissa Yadeta and S. Shaw, "Magnetic drug targeting during Casson blood flow in a microvessel: A caputo fractional model," *J. Magn. Magn. Mater.*, vol. 568, Feb. 2023, Art. no. 170363.

[21] E. P. Furlani, "Magnetic biotransport: Analysis and applications," *Materials*, vol. 3, no. 4, pp. 2412–2446, Mar. 2010.

[22] A. Nacev, C. Beni, O. Bruno, and B. Shapiro, "The behaviors of ferromagnetic nano-particles in and around blood vessels under applied magnetic fields," *J. Magn. Magn. Mater.*, vol. 323, no. 6, pp. 651–668, Mar. 2011.

[23] L. I. Kolitsi and S. G. Yiantsios, "Transport of nanoparticles in magnetic targeting: Comparison of magnetic, diffusive and convective forces and fluxes in the microvasculature, through vascular pores and across the interstitium," *Microvascular Res.*, vol. 130, Jul. 2020, Art. no. 104007.

[24] T. H. Boyer, "The force on a magnetic dipole," *Amer. J. Phys.*, vol. 56, no. 8, pp. 688–692, Aug. 1988.

[25] S. Yin, X. Li, H. Gao, and O. Kaynak, "Data-based techniques focused on modern industry: An overview," *IEEE Trans. Ind. Electron.*, vol. 62, no. 1, pp. 657–667, Jan. 2015.

[26] S. L. Brunton and J. N. Kutz, *Data-Driven Science and Engineering: Machine Learning, Dynamical Systems, and Control*. Cambridge, U.K.: Cambridge Univ. Press, 2022.

[27] R. Yu, S. L. Charreyron, Q. Boehler, C. Weibel, C. Chautems, C. C. Y. Poon, and B. J. Nelson, "Modeling electromagnetic navigation systems for medical applications using random forests and artificial neural networks," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2020, pp. 9251–9256.

[28] M. Cai, Q. Wang, Z. Qi, D. Jin, X. Wu, T. Xu, and L. Zhang, "Deep reinforcement learning framework-based flow rate rejection control of soft magnetic miniature robots," *IEEE Trans. Cybern.*, vol. 53, no. 12, pp. 7699–7711, Dec. 2022.

[29] L. Yang, J. Jiang, X. Gao, Q. Wang, Q. Dou, and L. Zhang, "Autonomous environment-adaptive microrobot swarm navigation enabled by deep learning-based real-time distribution planning," *Nature Mach. Intell.*, vol. 4, no. 5, pp. 480–493, May 2022.

[30] L. Xing, D. Li, H. Cao, L. Fan, L. Zheng, L. Zhang, and D. Sun, "A new drive system for microagent control in targeted therapy based on rotating gradient magnetic fields," *Adv. Intell. Syst.*, vol. 4, no. 9, Sep. 2022, Art. no. 2100214.

[31] M. D. Tehrani, J.-H. Yoon, M. O. Kim, and J. Yoon, "A novel scheme for nanoparticle steering in blood vessels using a functionalized magnetic field," *IEEE Trans. Biomed. Eng.*, vol. 62, no. 1, pp. 303–313, Jan. 2015.

[32] A. K. Hoshiar, T.-A. Le, P. Valdastri, and J. Yoon, "Swarm of magnetic nanoparticles steering in multi-bifurcation vessels under fluid flow," *J. Micro-Bio Robot.*, vol. 16, no. 2, pp. 137–145, Dec. 2020.

[33] T.-A. Le, M. P. Bui, and J. Yoon, "Electromagnetic actuation system for focused capturing of magnetic particles with a half of static saddle potential energy configuration," *IEEE Trans. Biomed. Eng.*, vol. 68, no. 3, pp. 869–880, Mar. 2021.

[34] A. Komaee and B. Shapiro, "Magnetic steering of a distributed ferrofluid spot towards a deep target with minimal spreading," in *Proc. 50th IEEE Conf. Decis. Control Eur. Control Conf.*, Dec. 2011, pp. 7950–7955.

[35] I. S. M. Khalil, J. D. Keuning, L. Abelmann, and S. Misra, "Wireless magnetic-based control of paramagnetic microparticles," in *Proc. 4th IEEE RAS EMBS Int. Conf. Biomed. Robot. Biomechatronics (BioRob)*, Jun. 2012, pp. 460–466.

[36] I. S. M. Khalil, L. Abelmann, and S. Misra, "Magnetic-based motion control of paramagnetic microparticles with disturbance compensation," *IEEE Trans. Magn.*, vol. 50, no. 10, pp. 1–10, Oct. 2014.

[37] R. Van Durme, G. Crevecoeur, L. Dupré, and A. Coene, "Model-based optimized steering and focusing of local magnetic particle concentrations for targeted drug delivery," *Drug Del.*, vol. 28, no. 1, pp. 63–76, Jan. 2021.

[38] R. Van Durme, G. Crevecoeur, L. Dupré, and A. Coene, "Improved magnetic drug targeting with maximized magnetic forces and limited particle spreading," *Med. Phys.*, vol. 50, no. 3, pp. 1715–1727, Mar. 2023.

[39] N. Talebloo, M. Gudi, N. Robertson, and P. Wang, "Magnetic particle imaging: Current applications in biomedical research," *J. Magn. Reson. Imag.*, vol. 51, no. 6, pp. 1659–1668, Jun. 2020.

[40] X. Han, K. Xu, O. Taratula, and K. Farsad, "Applications of nanoparticles in biomedical imaging," *Nanoscale*, vol. 11, no. 3, pp. 799–819, 2019.

[41] S. Xu, J. Liu, C. Yang, X. Wu, and T. Xu, "A learning-based stable servo control strategy using broad learning system applied for microrobotic control," *IEEE Trans. Cybern.*, vol. 52, no. 12, pp. 13727–13737, Dec. 2022.

[42] T.-A. Le, M. P. Bui, and J. Yoon, "Optimal design and implementation of a novel two-dimensional electromagnetic navigation system that allows focused heating of magnetic nanoparticles," *IEEE/ASME Trans. Mechatronics*, vol. 26, no. 1, pp. 551–562, Feb. 2021.

[43] M. P. Bui, T.-A. Le, and J. Yoon, "A magnetic particle imaging-based navigation platform for magnetic nanoparticles using interactive manipulation of a virtual field free point to ensure targeted drug delivery," *IEEE Trans. Ind. Electron.*, vol. 68, no. 12, pp. 12493–12503, Dec. 2021.

[44] O. Felfoul, A. T. Becker, G. Fagogenis, and P. E. Dupont, "Simultaneous steering and imaging of magnetic particles using MRI toward delivery of therapeutics," *Sci. Rep.*, vol. 6, no. 1, p. 33567, Sep. 2016.

[45] P. Benhal, A. Broda, D. Najafali, P. Malik, A. Mohammed, B. Ramaswamy, D. A. Depireux, M. Shimoji, M. Shukoor, and B. Shapiro, "On-chip testing of the speed of magnetic nano- and micro-particles under a calibrated magnetic gradient," *J. Magn. Magn. Mater.*, vol. 474, pp. 187–198, Mar. 2019.

[46] G. G. Stokes, *On the Effect of the Internal Friction of Fluids on the Motion of Pendulums*, vol. 9. Cambridge, U.K.: Pitt Press Cambridge, 1851.

[47] F. M. White, *Fluid Mechanics*, 7th ed. New York, NY, USA: McGraw-Hill, 2011.

[48] J. Carrey, B. Mehdaoui, and M. Respaud, "Simple models for dynamic hysteresis loop calculations of magnetic single-domain nanoparticles: Application to magnetic hyperthermia optimization," *J. Appl. Phys.*, vol. 109, no. 8, Apr. 2011, Art. no. 083921.

[49] G. Barrera, P. Allia, and P. Tiberto, "From spectral analysis to hysteresis loops: A breakthrough in the optimization of magnetic nanomaterials for bioapplications," *J. Phys., Mater.*, vol. 6, no. 3, Jul. 2023, Art. no. 035007.

[50] J.-B. Biot and F. Savart, "Note sur le magnétisme de la pile de Volta," *Ann. Chem. Phys*, vol. 15, p. 222, Jan. 1820.

[51] J. C. Simpson, J. E. Lane, C. D. Immer, and R. C. Youngquist, "Simple analytic expressions for the magnetic field of a circular current loop," NASA, Kennedy Space Center, Merritt Island, FL, USA, Tech. Rep. 20010038494, 2001.

[52] C. L. Lawson and R. J. Hanson, *Solving Least Squares Problems*. Philadelphia, PA, USA: SIAM, 1995.

[53] A. C. Müller and S. Guido, *Introduction to Machine Learning With Python: A Guide for Data Scientists*. Sebastopol, CA, USA: O'Reilly Media, 2016.

[54] M. A. Nielsen, *Neural Networks and Deep Learning*, vol. 25. San Francisco, CA, USA: Determination Press, 2015.

[55] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. Cambridge, MA, USA: MIT Press, 2016. [Online]. Available: http://www.deeplearningbook.org

[56] C. M. Bishop, "Training with noise is equivalent to Tikhonov regularization," *Neural Comput.*, vol. 7, no. 1, pp. 108–116, Jan. 1995.

[57] H. Jiang, *Machine Learning Fundamentals: A Concise Introduction*. Cambridge, U.K.: Cambridge Univ. Press, 2021.

[58] H. Antil, R. H. Nochetto, and P. Venegas, "Optimizing the Kelvin force in a moving target subdomain," *Math. Models Methods Appl. Sci.*, vol. 28, no. 1, pp. 95–130, Jan. 2018.

[59] M. Diehl, H. G. Bock, H. Diedam, and P.-B. Wieber, "Fast direct multiple shooting algorithms for optimal robot control," in *Fast Motions in Biomechanics and Robotics*. Cham, Switzerland: Springer, 2006, pp. 65–93.

[60] J. Nocedal and S. Wright, *Numerical Optimization*. Berlin, Germany: Springer, 2006.

[61] B. Gleich and J. Weizenecker, "Tomographic imaging using the nonlinear response of magnetic particles," *Nature*, vol. 435, no. 7046, pp. 1214–1217, Jun. 2005.

[62] T.-A. Le, X. Zhang, A. K. Hoshiar, and J. Yoon, "Real-time two-dimensional magnetic particle imaging for electromagnetic navigation in targeted drug delivery," *Sensors*, vol. 17, no. 9, p. 2050, Sep. 2017.

[63] F. Griese, T. Knopp, C. Gruettner, F. Thieben, K. Müller, S. Loges, P. Ludewig, and N. Gdaniec, "Simultaneous magnetic particle imaging and navigation of large superparamagnetic nanoparticles in bifurcation flow experiments," *J. Magn. Magn. Mater.*, vol. 498, Mar. 2020, Art. no. 166206.

[64] K. Belharet, D. Folio, and A. Ferreira, "Real-time software platform using MRI for in vivo navigation of magnetic microrobots," in *Medical Robotics*. Amsterdam, The Netherlands: Elsevier, 2012, pp. 252–275.
[65] C. Van Heck, A. Coene, and G. Crevecoeur, "Preventing catastrophic forgetting using prior transfer in physics informed Bayesian neural networks," in *Proc. IEEE/ASME Int. Conf. Adv. Intell. Mechatronics (AIM)*, Jul. 2022, pp. 650–657.
[66] E. F. Camacho and C. B. Alba, *Model Predictive Control*. Berlin, Germany: Springer, 2013.

RIKKERT VAN DURME was born in 1994. He received the M.Sc. degree in electromechanical engineering from Ghent University, Belgium, in 2017, where he is currently pursuing the Ph.D. degree with the Department of Electromechanical, Systems and Metal Engineering. In September 2017, he joined the Department of Electromechanical, Systems and Metal Engineering, Ghent University, as a Teaching Assistant. His research interest includes the modeling, optimization, and control of magnetic nano- and microparticles for biomedical applications.

GUILLAUME CREVECOEUR (Member, IEEE) was born in 1981. He received the M.Sc. and Ph.D. degrees in engineering physics from Ghent University, Belgium, in 2004 and 2009, respectively. In 2009, he became a Postdoctoral Fellow with the Research Foundation–Flanders (FWO). In 2014, he was appointed as an Associate Professor with the Faculty of Engineering and Architecture, Ghent University. His research interests include the modeling, optimization, and control of dynamical mechatronic and industrial robotic systems.

LUC DUPRÉ (Member, IEEE) was born in 1966. He received the degree in electrical and mechanical engineering and the Ph.D. degree in applied sciences from Ghent University, Belgium, in 1989 and 1995, respectively. In 1989, he joined the Department of Electrical Energy, Systems and Automation, Ghent University, as a Research Assistant. In 1996, he became a Postdoctoral Fellow with the Research Foundation–Flanders (FWO). In 2002, he was appointed as an Associate Professor with the Faculty of Engineering and Architecture, Ghent University, where he has been a Full Professor, since 2006. His research interests include numerical methods for low-frequency electromagnetism, modeling, and characterization of magnetic materials, micromagnetism, and electromagnetism for biomedical applications.

ANNELIES COENE was born in 1988. She received the M.Sc. and Ph.D. degrees in biomedical engineering from Ghent University, Belgium, in 2011 and 2017, respectively. In 2017, she became a Postdoctoral Fellow with the Research Foundation–Flanders (FWO). Her research interests include the merging of physics-based models with data-driven approaches to improve, monitor, and predict the performance of biomedical, and mechatronic applications.