

# Efficient Model Learning and Adaptive Tracking Control of Magnetic Micro-Robots for Non-Contact Manipulation

Yongyi Jia, Shu Miao, Junjian Zhou, Niandong Jiao, Lianqing Liu, and Xiang Li

**Abstract**—Magnetic microrobots can be navigated by an external magnetic field to autonomously move within living organisms with complex and unstructured environments. Potential applications include drug delivery, diagnostics, and therapeutic interventions. Existing techniques commonly impart magnetic properties to the target object, or drive the robot to contact and then manipulate the object, both probably inducing physical damage. This paper considers a non-contact formulation, where the robot spins to generate a repulsive field to push the object without physical contact. Under such a formulation, the main challenge is that the motion model between the input of the magnetic field and the output velocity of the target object is commonly unknown and difficult to analyze. To deal with it, this paper proposes a data-driven solution. A neural network is constructed to efficiently estimate the motion model. Then, an approximate model-based optimal control scheme is developed to push the object to track a time-varying trajectory, maintaining the non-contact with distance constraints. Furthermore, a straightforward planner is introduced to assess the adaptability of non-contact manipulation in a cluttered unstructured environment. Experimental results are presented to show the tracking and navigation performance of the proposed scheme.

## I. INTRODUCTION

Robotics micromanipulation [1], [2], [3] is a prominent and active research area, which holds paramount significance across an array of applications within disciplines, including biomedicine, electronics, and materials science [4], [5]. Many strategies have been applied to micromanipulation automation, such as mechanical [6], fluidic [7], electrical [8], optical [9] and magnetic [10]. Among them, magnetic microrobots have the attractive feature of moving within living organisms. To improve its autonomous capability, diverse methodologies have been devised to attain meticulous command over the positioning, orientation, and motion of magnetic particles [11], [12], [13]. However, existing results are commonly under the formulation of direct manipulation [14], imparting magnetic properties to the target object, which would disrupt its inherent properties, or contact manipulation, driving the robot to contact and then manipulate the object [15]. The introduction of physical contact would probably result in deformation, contamination, and difficulty in separating. To deal with the

Y. Jia, S. Miao, and X. Li are with the Department of Automation, Tsinghua University, China. J. Zhou, N. Jiao and L. Liu are with the State Key Laboratory of Robotics, Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China. This work was supported in part by Tsinghua University Initiative Scientific Research Program, in part by Beijing National Research Center for Information Science and Technology, in part by the National Natural Science Foundation of China under Grant U21A20517 and 52075290, and in part by the Science and Technology Innovation 2030-Key Project under Grant 2021ZD0201404. Corresponding author: Xiang Li (xiangli@tsinghua.edu.cn)

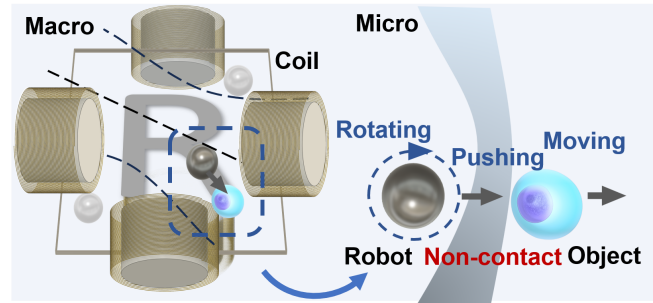


Fig. 1. Illustration diagram about magnetic robot performing non-contact manipulation on the target object. The rotating magnetic field generated by the electromagnetic coil propels the robot to roll, thereby inducing fluid motion to drive the target object.

forementioned drawbacks, this paper will be dedicated to proposing an indirect control approach that does not involve contact. Basically, the robot rotates under a magnetic field to generate repulsive and eddy fields to push the object, as shown in Fig. 1. Under such a formulation, the analysis of the motion model from magnetic field input to target velocity output is a complex fluid-structure interaction (FSI) problem that involves the bidirectional influence between the robot, the target object, and the surrounding fluid. Furthermore, the uncertainty of dynamic parameters adds challenge to the acquisition of the motion model.

This paper introduces a data-driven solution to the aforementioned problem. While much data-driven progress has been reported to estimate the unknown model at the macro scale, it is not trivial to reproduce it at the micro scale since i) few sensory techniques, ii) stochastic Brownian motion, and especially, iii) the inefficiency or even absence of simulation methods making data collection exceedingly challenging.

In this paper, a neural network is constructed to quickly estimate the proposed succinct model expression with a real-world collected small data set. Then, a model-based control scheme is developed to push the target object, tracking the desired time-varying trajectory. The control input is obtained by solving a convex optimization problem with distance constraints, guaranteeing real-time performance and non-contact ability. Finally, a curvature optimization-based planner is introduced to validate the navigation ability of non-contact manipulation. Experimental results are carried out to show the tracking and navigation performance of the proposed method.

The indirect non-contact manipulation manner is similar to the non-prehensile manipulation [16], especially planar pushing [17], [18] at the macro scale, such that many newly-emerging techniques at the macro scale can also be applied to the microrobot to bridge the gap between each other.

## II. RELATED WORKS

Magnetic field is most promising for the cell transport function due to its high penetration of body, precise controllability and good biocompatibility [19]. Several works have been proposed to manipulate micro-objects in a non-contact manner under a magnetic and flow field. One approach involves propelling the robot translationally using a gradient magnetic field, thereby generating laminar flow along the robot's boundaries. Floyd et al. [20] introduced a method that employs micro-robots to harness the fluid velocity field generated by the fluid boundary layer, thereby inducing the non-contact motion of the target object. Then it was expanded by Pawashe et al. in [21] as two manipulation strategies: 1) front pushing, involving direct contact between the microrobot and the micro-object; and 2) side pushing, which enables non-contact displacement through the utilization of fluid flow fields generated by a translating microrobot. The effective range of the translational mode has limited effectiveness, making long-distance transportation of objects challenging to achieve.

Another approach involves leveraging the eddy currents generated by the robot's rotating or rolling in a uniform magnetic field. Pieters et al. [22] presented the modeling and control of a rolling microrobot, the RodBot. The RodBot is capable of executing non-contact pushing operations and lifting objects utilizing a vortex above its body. Petit et al. [23] introduced a novel approach for achieving controlled manipulation of individual micro- and nanoscale objects through the utilization of mobile micro vortices, which enabled trapping and manipulation within fluidic environments without significant constraints on material properties. Ye et al. [24] introduced a technique that demonstrated the ability to dynamically trap and precisely transport individual swimming microorganisms in a two-dimensional manner, all without requiring the use of additional labeling on the biological samples or specialized surface patterns. Lin et al. [25] presented a magnetically actuated peanut-shaped hematite colloid motor capable of both rolling and wobbling motion in fluids, which enabled single-cell non-contact transportation. This eddy non-contact mode is more flexible while achieving closed-loop control using eddy can be challenging. Therefore, Khalil et al. [26] proposed a non-contact manipulation strategy using a rotating two-tailed soft microrobot to induce a frequency-dependent flow-field for orbiting nonmagnetic microbeads, with a closed-loop controller to guide the microbeads towards target positions.

Our work is inspired by recent research [27], which introduced a conceptual diagram along with experimental demonstrations depicting a non-contact object manipulation scenario by the cell robot. Specifically, during rotation, hydrodynamic pressure at the robot's edge effectively pushes the object forward. Compared to the previous two non-contact methods, this approach utilizing local pressure proves to be more stable and easier to control. Therefore, we employ this non-contact mode based on pushing, enabling precise closed-loop control and automatic navigation.

In summary, non-contact indirect manipulation offers

advantages such as the absence of physical damage, reduced contamination risk, and ease of separation, which have garnered the attention of researchers. However, there are relatively few works regarding the closed-loop control and navigation of non-contact manipulation, which are either controlled by humans or applicable to setpoint control only, limiting its applications.

## III. INDIRECT NON-CONTACT MANIPULATION

In this section, we explore models of non-contact indirect magnetic drive systems. Section III. A presents the principle and steady-state analysis of the one-dimensional scenario, illustrating how magnetic robots manipulate objects without contact. Section III. B extends this to two-dimensional motion, introducing dynamic equations for the following control.

### A. Linear Non-Contact Motion

As illustrated in Fig. 2(a), a spherical robot with rolling motion is driven by a rotating magnetic field in the x-z plane. A distinctive rotating uniform magnetic field is characterized by an angular velocity  $\omega_b$  and an amplitude  $B$ , expressed as

$$\mathbf{B}(\omega_b, t) = B \cos(\omega_b t + \theta) \mathbf{e}_x + B \sin(\omega_b t + \theta) \mathbf{e}_z, \quad (1)$$

where  $\mathbf{e}_x$  and  $\mathbf{e}_z$  denote the unit vectors along the x-axis and z-axis, respectively, and  $\theta$  represents the initial phase angle. Due to the non-uniform magnetic sputtering on the surface, the robot microsphere has a magnetic moment  $\mathbf{m}$  and then is subjected to a magnetic torque, which can be represented as

$$\tau_m = \mathbf{m} \times \mathbf{B} = mB \sin(\Phi), \quad (2)$$

where  $\Phi$  is the angle between the magnetic moment and the magnetic field. The equilibrium between magnetic torque and viscous torque infers that the robot rotates at the same angular velocity as the magnetic field, with a linear velocity proportional to the angular velocity, that is  $\omega_r = \omega_b$ , and  $v_r = a\omega_b$ , until the step-out frequency is reached [28].

When focusing on indirect manipulation, the rotation of the robot drives surrounding fluid flow with a surface velocity as  $\omega_r r$ , where  $r$  is the radius of the robot, significantly outpacing the robot's linear speed  $v_r$ , which results in high-pressure ahead the robot, facilitating conditions for indirect non-contact manipulation of objects [27]. Considering both the robot and target object as a single entity, the rotation and linear motions are obtained as

$$\tau_m - C_m^r \omega_r - F_d r = 0, \quad (3)$$

$$F_d - C_d^r v_r - C_d^o v_o = 0, \quad (4)$$

where  $v_o$  is the linear velocity of the target object,  $F_d$  is the drag force between the robot and the underlying surface [29],  $C_m$  is the viscous moment coefficient, and  $C_d$  is the viscous drag coefficient. As the distance between them diminishes, pressure increases until the target object's speed matches the robot's linear speed, that is,  $v_r = v_o$ , resulting in a constant phase angle difference  $\Phi$  and a relative state of rest. The ratio of sliding and rotating  $\frac{v_r}{\omega_r}$  which depends on both object property and environmental damping can be obtained by measuring the linear velocity.

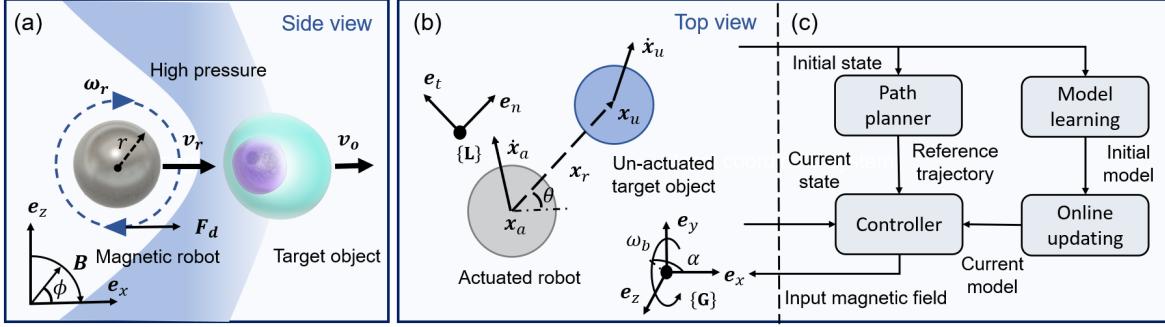


Fig. 2. Overview of the proposed non-contact manipulation scheme. (a) One-dimensional schematic diagram of non-contact pushing. (b) Two-dimensional motion model employing a local coordinate system, where the normal and the tangential are coupled. (c) Control structure of non-contact manipulation. The controller receives the reference trajectory from the planner and the estimation model based on the neural network, then gives the control input through optimization and online updating.

### B. Planar Non-Contact Motion

In this paper, we focus on addressing the task of the magnetic-driven robot: 1) pushing an object along a designated trajectory and 2) navigating it to a target location in a cluttered environment within a two-dimensional plane. Both the robot and the manipulated object are assumed to be homogeneous spheres suitable for most microscopic manipulation targets.

As shown in Fig. 2(b), the magnetic field rotates around an axis in the  $xy$  plane. The angle between its projection onto the  $xy$  plane and the  $x$ -axis is denoted as  $\alpha$ . When the robot's motion direction differs from the line connecting it to the object's center, a tangential fluid force affects the object, causing its curved motion. Due to the intricate dynamics of fluid-structure interaction [30], calculating these challenges remains tough, even using a multiphysics simulator such as COMSOL Multiphysics [31]. Hence, we introduce a control affine model to depict the planar motion, which can be estimated through a data-driven method.

Below, the positions of the actuated robot and un-actuated target object are denoted as  $\mathbf{x}_a \in \mathbb{R}^2$  and  $\mathbf{x}_u \in \mathbb{R}^2$ . Since we are only concerned with the control of the target object while maintaining a non-contact state between the two objects, the system state is set to  $\mathbf{x} = [\mathbf{x}_u, \mathbf{x}_r]^T \in \mathbb{R}^4$ , where  $\mathbf{x}_r = \mathbf{x}_u - \mathbf{x}_a$  is the relative position. The control input  $\mathbf{u}$  is defined as  $[\omega_b \cos \alpha, \omega_b \sin \alpha]^T \in \mathbb{R}^2$ , representing the projected angular velocity on the  $x$  and  $y$ -axis. Several properties can be noticed intuitively: 1) quasi-static: under low Reynolds coefficients, viscous resistance plays the prevailing factor and the system follows a first-order model; 2) Movement is solely induced by a magnetic field, rendering the drift vector field  $f(\cdot)$  of the system as none; 3) Both the liquid and magnetic field are uniformly distributed, then the control vector field  $g(\cdot)$  is solely determined by relative positioning. Therefore, the system model can now be given as

$$\dot{\mathbf{x}} = \mathbf{g}(\mathbf{x}_r)\mathbf{u}, \quad (5)$$

where  $\mathbf{g} : \mathbb{R}^2 \rightarrow \mathbb{R}^{4,2}$  is the unknown nonlinear control function about  $\mathbf{x}_r$ .

## IV. MODELING LEARNING AND TRACKING CONTROL

This section proposes a new adaptive tracking controller to achieve the non-contact pushing task. First, a neural network

is constructed to efficiently estimate the motion model. The controller is formulated as an optimization problem, such that constraints can also be added to avoid contact. A planner with curvature optimization is proposed to verify the navigation capability. The whole structure of the scheme is illustrated in Fig. 2(c).

### A. Model Learning

An offline data-driven strategy is adapted for constructing the dynamic model to overcome the analytical challenges caused by fluid-structure interaction. To enhance model simplicity and reduce the data sampling requirements, we introduce a local coordinate system, serving the line connecting the robot's center and the target object's center as the normal direction reference, as shown in Fig. 2(b). The control equation exclusively depends on the relative distance. Additionally, the motion along the tangential direction and the normal direction becomes decoupled. To further improve model generalization, we scale-normalize the relative distances by the mean radius of the robot and the object.

Then, (5) can be expressed in concise forms

$$\dot{\mathbf{x}}_u = \mathbf{g}_u(\mathbf{x}_r)\mathbf{u} = \mathbf{R}^T \mathbf{N}_u(s_r)\mathbf{R}\mathbf{u}, \quad (6)$$

$$\dot{\mathbf{x}}_r = \mathbf{g}_r(\mathbf{x}_r)\mathbf{u} = \mathbf{R}^T \mathbf{N}_r(s_r)\mathbf{R}\mathbf{u}, \quad (7)$$

where  $\mathbf{R} \in \mathbb{SO}(2)$  is the global-to-local rotation matrix for the angle  $\arctan(\mathbf{x}_{rx}, \mathbf{x}_{ry})$ ,  $s_r \in \mathbb{R}$  is shorthand for normalized relative Euclidean distance  $\|\mathbf{x}_r\|_2 / (r_a + r_u)$ , and  $\mathbf{N}_u, \mathbf{N}_r : \mathbb{R} \rightarrow \mathbb{R}^{2,2}$  return diagonal matrices due to direction decoupling. A unified form is used to represent the following unactuated and relative formulas for simplicity, distinguishing by subscript  $k \in \{u, r\}$ .

We apply radial basis function network (RBFN) to estimate the unknown nonlinear functions  $\mathbf{N}_u$  and  $\mathbf{N}_r$ . RBFN possesses the capability for rapid learning and can be updated online [32], attaining significant advancements in the modeling and control of deformable objects [33]. The actual function is expressed as

$$\mathbf{N}_k(s_r) = \text{diag}(\mathbf{W}_k \boldsymbol{\phi}(s_r)), \quad k \in \{u, r\} \quad (8)$$

where  $\boldsymbol{\phi}(s_r) = [\phi_1(s_r), \phi_2(s_r), \dots, \phi_p(s_r)]^T \in \mathbb{R}^p$  is the activation function vector shared between two networks. Each network is equipped with its own set of weights,  $\mathbf{W}_u$  and

$\mathbf{W}_r \in \mathbb{R}^{2,p}$ , and  $\text{diag}(\cdot)$  represents the operator creating diagonal matrices from vectors. Given micro-environmental motion with Brownian noise, we employ a multi-quadric radial function as the activation function.

$$\phi_i(s_r) = \sqrt{1 + \|\sigma_i(s_r - \mu_i)\|_2^2}. \quad i = 1, 2, \dots, p \quad (9)$$

The parameters  $\mu_i$  and  $\sigma_i$  are the center and width of the  $i$ th neuron, which are trained offline and fixed online. This activate function could maintain smoothness in input parameters while offering robustness in noisy data scenarios.

By substituting (8) to (6) and (7), the actual motion model is rewritten as

$$\dot{\mathbf{x}}_k = \mathbf{R}^\top \text{diag}(\mathbf{W}_k \phi(s_r)) \mathbf{R} \mathbf{u}. \quad (10)$$

With the estimated weight matrix  $\hat{\mathbf{W}}_k$  trained from the data set, the estimated function is represented as

$$\hat{\mathbf{g}}_k(\mathbf{x}_r) = \mathbf{R}^\top \text{diag}(\hat{\mathbf{W}}_k \phi(s_r)) \mathbf{R}. \quad (11)$$

Thus, the prediction error of the model is calculated as

$$e_k = \dot{\mathbf{x}}_k - \hat{\mathbf{g}}_k(\mathbf{x}_r) \mathbf{u} = \mathbf{R}^\top \text{diag}(\Delta \mathbf{W}_k \phi(s_r)) \mathbf{R} \mathbf{u}, \quad (12)$$

where  $\Delta \mathbf{W}_k = \mathbf{W}_k - \hat{\mathbf{W}}_k$  is the approximation error of the RBFN weights. We manually manipulate the magnetic field's direction and angular velocity in a real environment to collect offline data. Each data tuple encompasses relative position, object velocity, relative velocity, and magnetic field input, as  $\{\mathbf{x}_r, \mathbf{v}_u, \mathbf{v}_r, \mathbf{u}\}$ . The data set is ensured to span a broad range of relative distances from contacting to keeping away. Mean squared error (MSE) loss of  $e_k$  and Adam optimizer [34] are used for offline training. The simplicity of the RBFN's structure enables excellent fitting performance with a minimal data requirement, thus facilitating manual data collection.

### B. Adaptive Optimal Control

After estimating the nonlinear system model, a straightforward approach involves employing feedback linearization for control. However, it is crucial to ensure that the robot maintains a safe distance from the target object to facilitate non-contact. To address this challenge, we introduce an adaptive optimal controller. A convex optimization problem is formulated to compute the control inputs while incorporating distance constraints to prevent any physical contact. Simultaneously, the controller dynamically updates the neural network weights in real time to compensate for modeling errors, thus enabling adaptation to alterations in both objects and environmental conditions.

The trajectory tracking is transformed into a series of fixed-point control problems involving a sequence of target points. The desired target position  $\mathbf{x}_u^{\text{des}}$  is chosen as the next point on the trajectory, and then the target task error is defined as

$$\Delta \mathbf{x}_u = \mathbf{x}_u - \mathbf{x}_u^{\text{des}}. \quad (13)$$

To push the target object forward along the trajectory, the desired relative position is aligned opposite to the initial target task error as

$$\mathbf{x}_r^{\text{des}} = -x_r^{\text{avg}} \frac{\mathbf{x}_u - \mathbf{x}_u^{\text{des}}}{\|\mathbf{x}_u - \mathbf{x}_u^{\text{des}}\|_2}, \quad (14)$$

where  $x_r^{\text{avg}}$  is the average value of the relative distance in the data set corresponding to a specific range of target object speeds. The relative task error is obtained as  $\Delta \mathbf{x}_r = \mathbf{x}_r - \mathbf{x}_r^{\text{des}}$ . Then, ideal velocity vectors are defined in the opposite direction of the task errors as

$$\dot{\mathbf{x}}_u^{\text{ide}} = -\alpha \Delta \mathbf{x}_u, \quad (15)$$

$$\dot{\mathbf{x}}_r^{\text{ide}} = -\alpha \Delta \mathbf{x}_r, \quad (16)$$

where  $\alpha$  is a positive gain factor.

The control input is obtained by solving the following convex optimization problem

$$\begin{aligned} \min_{\mathbf{u}} \quad & \frac{1}{2} \|\dot{\mathbf{x}}_u^{\text{ide}} - \hat{\mathbf{g}}_u(\mathbf{x}_r) \mathbf{u}\|_2^2 + \frac{\lambda}{2} \|\dot{\mathbf{x}}_r^{\text{ide}} - \hat{\mathbf{g}}_r(\mathbf{x}_r) \mathbf{u}\|_2^2 \\ \text{s.t.} \quad & \|\mathbf{u}\|_2 \leq u_{\max} \\ & \mathbf{x}_r^\top \hat{\mathbf{g}}_r(\mathbf{x}_r) \mathbf{u} \leq 0 \quad \text{if } s_r \leq s_r^{\min} \\ & \mathbf{x}_r^\top \hat{\mathbf{g}}_r(\mathbf{x}_r) \mathbf{u} \geq 0 \quad \text{if } s_r \geq s_r^{\max} \end{aligned} \quad (17)$$

where the cost function is designed to bring the target object closer to the desired position as possible, and  $\lambda$  is a tunable parameter whose increase aids in tracking trajectories with higher curvature. As mentioned above, the contact between the robot and the target object will cause solid contamination, and it is difficult to separate due to the adhesion force. Therefore, we introduce safety distance constraints to maintain a non-contact state. As indicated in the second inequality, when the distance is less than or equal to the minimum safe distance  $s_r^{\min}$ , the projection of the relative velocity onto the relative distance becomes negative, resulting in an expansion of the relative distance. This principle also applies when the distance exceeds the maximum operating distance  $s_r^{\max}$ .

The online updating law of the estimated weight matrix is designed as

$$\begin{aligned} \dot{\hat{\mathbf{W}}}_{k,i}^\top &= \lambda_k \sum_{j=1}^2 R_{i,j} (\tau_1 \Delta x_{k,j} + \tau_2 e_{k,j}) \mathbf{R}_i \mathbf{u} \phi(s_r), \\ i = 1, 2, \quad \lambda_k &= \begin{cases} 1, & k = u \\ \lambda, & k = r \end{cases} \end{aligned} \quad (18)$$

where  $\Delta x_{k,j}$  are the  $j$ th element of the target and relative task error,  $e_{k,j}$  are the  $j$ th element of the target and relative prediction error, and  $\tau_1, \tau_2$  are positive scalars. The control rate (17) combined with the online update rate (18) theoretically guarantees the Lyapunov stability of the closed-loop system [35].

### C. Navigation in Clutter Environments

We develop a straightforward path planning method for guiding a target object to its destination, demonstrating the application of our proposed automated non-contact manipulation in real clutter micro-environments. Addressing the curvature limitations imposed by non-holonomic constraints on the target object's trajectory, we propose the Curvature-Optimized RRT (CO-RRT) algorithm, an enhancement to the Rapidly-exploring Random Tree (RRT) framework.

First, the turning angle is considered in the distance calculation when checking the nearest node of the randomly

sampled one. The turning angle is defined as

$$\theta = \arccos \left( \frac{\mathbf{x}_c \cdot \mathbf{x}_p}{\|\mathbf{x}_c\|_2 \|\mathbf{x}_p\|_2} \right), \quad (19)$$

where  $\mathbf{x}_c$  is the position of current node and  $\mathbf{x}_p$  is the position of parent node. The distance measure is designed by the summation of the orientation difference and the Euclidean distance as

$$d(\mathbf{p}_r, \mathbf{p}_t) = \left| \arccos \left( \frac{\text{tr}(\mathbf{R}_r^T \mathbf{R}_t)}{2} \right) \right| + \|\mathbf{x}_r - \mathbf{x}_t\|_2, \quad (20)$$

where the subscripts  $r, t$  respectively represent randomly sampled nodes and nodes on the expanding tree.  $\mathbf{R}_r$  and  $\mathbf{R}_t$  are the rotation matrices of turning angles  $\theta_r$  and  $\theta_t$ .

After obtaining a set of reference trajectory points  $\mathbf{x}_1^{\text{des}}, \mathbf{x}_2^{\text{des}}, \dots, \mathbf{x}_n^{\text{des}}$ , we employ Bézier curves [36] to smooth the trajectory. The continuous reference trajectory along the Bézier curve is depicted as

$$\mathbf{x}^{\text{des}}(t) = \sum_{i=0}^n b_i^q(t) \mathbf{x}_i, \quad (21)$$

where  $b_i^q(t)$  are the Bézier basis functions defined as

$$b_i^q(t) = \frac{q!}{i!(q-i)!} (1-t)^{q-i} t^i. \quad (22)$$

The degree  $q$  of the Bezier curve determines its smoothness and complexity. This approach diminishes the maximum curvature of the planned trajectory, consequently leading to a substantial reduction in tracking errors.

## V. EXPERIMENTS

As shown in Fig. 3, the system consists of a self-designed three-axis Helmholtz electromagnetic coil mounted on the motorized stage of the inverted microscope (Nikon, ECLIPSE Ti2-E), in which the specimen is embedded via a 3D printed support structure. A CMOS camera (HIKROBOT, MV-CH050-10UM) is connected to the inverted microscope to capture images. On the PC side, OpenCV [37] is employed for the recognition and localization of the magnetic robot and target object. Subsequently, control inputs are computed and transmitted to a servo amplifier (Maxon, ESCON 50/5) to drive the electromagnetic coil. The amplitude of the magnetic field is maintained at 10 mT, and the angular velocity and direction of rotation are calculated by the controller. The magnetic robot is fabricated by spin-coating a soft magnetic nickel layer onto silica microspheres. Using 10  $\mu\text{m}$  microspheres, one hour of DC magnetron sputtering results in a nickel coating thickness of approximately 480 nm. The target objects consist of untreated 10  $\mu\text{m}$  microspheres.

### A. Validation of Model Learning Method

We first present a series of comparative studies illustrating the efficiency and real-time performance of the proposed model estimation method. Offline data is collected manually using Labview in the real environment. Subsequently, the RBF network approximation model is trained using the method

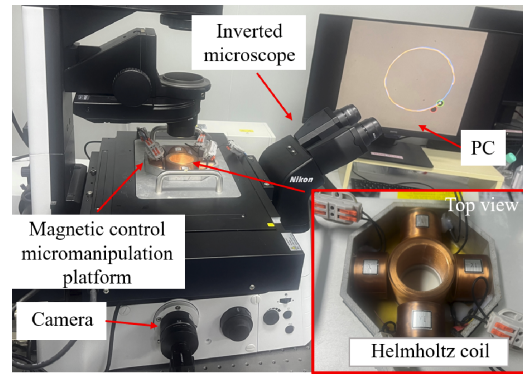


Fig. 3. System setup. The three-axis Helmholtz coil generates a rotating magnetic field, while a camera is connected to an inverted microscope to capture images. The magnetic field inputs are calculated on the PC side.

TABLE I  
ACTIVATION FUNCTIONS AND NEURON NUMBERS

Neurons number		8	16	32	64
$e_u^{\text{test}} \downarrow$	Multi-quadric	0.192	0.198	<b>0.186</b>	0.228
	Gaussian	0.192	<b>0.186</b>	0.188	/ <sup>1</sup>
$e_r^{\text{test}} \downarrow$	Multi-quadric	0.222	0.226	<b>0.213</b>	0.222
	Gaussian	0.219	0.215	0.221	0.214

<sup>1</sup> Gradient vanishing during the training process.

outlined in Section IV-A. The relative prediction errors of  $\dot{\mathbf{x}}_u$  and  $\dot{\mathbf{x}}_r$  are defined as

$$e_k^{\text{test}} = \frac{\|\dot{\mathbf{x}}_k - \hat{\mathbf{g}}_k(\mathbf{x}_r) \mathbf{u}\|_2}{\|\dot{\mathbf{x}}_k\|_2} \times 100\%. \quad k \in \{u, r\} \quad (23)$$

First, we evaluate the impact of different models and the amount of training data on the modeling performance. As shown in Fig. 4, the model corresponding to (5) exhibits the highest test error due to the absence of local coordinate system projection. Our model utilizing the decoupling of normal and tangential directions demonstrates superior data efficiency compared to the non-decoupled model. The relative modeling error under the training set of about 800 is about 20%, among the same order of magnitude as the Brownian motions (with an average displacement of 20  $\mu\text{m}$  per second according to the Stokes-Einstein equation), validating the efficiency of our modeling approach.

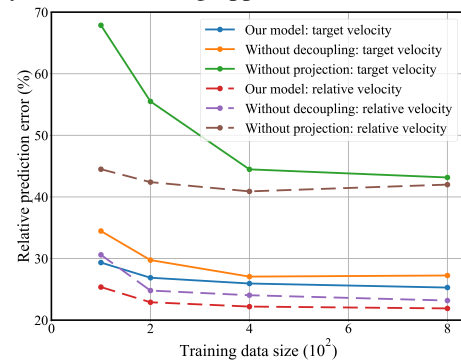


Fig. 4. Validation of the data efficiency of local coordinate projection and decoupling method

We then compare the impact of two activation functions and the number of hidden layers of neurons on modeling accuracy. As summarized in Table I, the lowest prediction

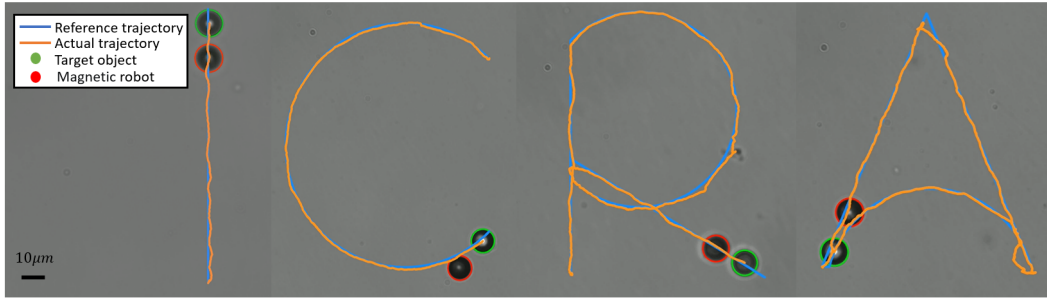


Fig. 5. Trajectory tracking of “ICRA”. The green circle represents the target object, the red circle represents the magnetic-driven robot, the blue line represents the reference trajectory, and the orange line represents the actual trajectory.

error is achieved with 32 neurons, ensuring the suitability of RBFN for real-time control. The multi-quadric activation function is ultimately chosen for its stability during training and smoothness of input.

### B. Validation of Trajectory Tracking Method

To validate the dexterity of non-contact manipulation and the effectiveness of the proposed control method, we first designed trajectories involving the tracking of the four letters “ICRA”, which includes both straight-line segments and curves with different curvatures. The hyperparameters for the controller are set as  $\alpha = 1.0$ ,  $\lambda = 2.0$ ,  $u_{\max} = 2\pi(\text{rad/s})$ ,  $s_r^{\min} = 2.25$ , and  $s_r^{\max} = 4$ , those for the online updater are  $\tau_1 = \tau_2 = 0.1$ . As shown in Fig. 5, the controller can successfully complete the tracking of all four trajectories while maintaining a non-contact state. The average tracking error for the letter “C” is minimized, approximately  $0.4 \mu\text{m}$ , with the maximum error occurring at the non-differentiable corner of the letter “A”, measuring  $7.3 \mu\text{m}$ .

We compare our proposed model-based optimal controller with the model-free Proportional (P-) controller. The actual trajectories are shown in Fig. 6. Our method completes the task. However, the error of the P-controller fails to converge, further, the robot and the object come into contact at  $[31.82\mu\text{m}, 41.80\mu\text{m}]$ , leading to adhesion and ultimately resulting in task failure.

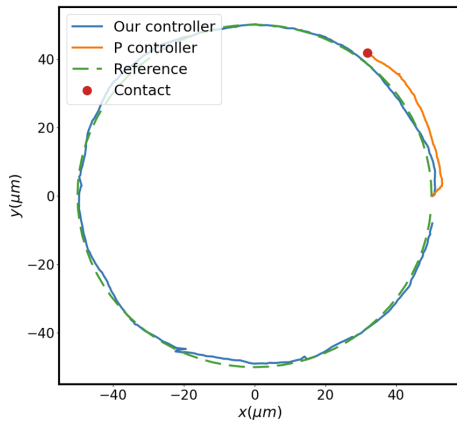


Fig. 6. Comparison of the model-based optimal controller with proportional controller. The proportional controller leads to task failure since the robot comes into contact with the target object.

We further investigate the influence of control parameter  $\lambda$  and curvature on tracking error for circular trajectories. Curvature is calculated by  $\frac{1}{R}$ . The average tracking error

TABLE II  
CONTROL PARAMETERS AND TRAJECTORY CURVATURES

Mean relative error (100%)↓	$\lambda$			
	1	2	5	
Radius ( $\mu\text{m}$ )	25	5.77	2.39	<b>1.83</b>
	50	0.626	<b>0.378</b>	0.409
	75	0.192	<b>0.186</b>	0.188

TABLE III  
COMPARISON BETWEEN DIFFERENT PLANNERS

Method	Maximum curvature ↓ (rad/pixel)	Maximum tracking ↓ error ( $\mu\text{m}$ ) <sup>1</sup>
CO-RRT	<b>0.0084</b>	<b>2.654</b>
Non-smooth	0.0532	5.320
RRT	0.0827	14.367

<sup>1</sup> Due to the real-time variability of the experimental environment, tracking errors are tested under uniform non-obstacle conditions.

is recorded as TABLE II. As the curvature increases, the relative tracking error increases, and the coefficient  $\lambda$  should accordingly increase to achieve better tracking performance.

Finally, we validate the effectiveness of navigation in cluttered scenarios by comparing our planner with the planner without smoothing and the original RRT method. As shown in TABLE III, our method generates a trajectory with the highest curvature, leading to the smallest tracking error. Additional navigation results are provided in the supplementary video.

## VI. CONCLUSIONS

This work presents a novel model, control, and navigation strategies for magnetic-driven non-contact micromanipulation. We introduce a decoupled affine non-linear model to delineate the dynamics of fluid-force effection subject to external magnetic field input. Subsequently, the neural networks are employed to efficiently learn the model from data and update weights online to enhance convergence speed. We propose an optimal controller based on the aforementioned model, which incorporates constraints to ensure non-contact manipulation. Finally, a curvature optimization planner is employed to guarantee navigation in clutter scenarios. The application of these techniques allows for precise control of target objects in a non-contact manner, facilitating trajectory tracking and reaching specific positions, without causing any physical damage. Future work will focus on applying this technology to single-cell manipulation.

## REFERENCES

- [1] J. Rufo, F. Cai, J. Friend, M. Wiklund, and T. J. Huang, "Acoustofluidics for biomedical applications," *Nature Reviews Methods Primers*, vol. 2, no. 1, p. 30, 2022.
- [2] X. Wang, C. Ho, Y. Tsatskis, J. Law, Z. Zhang, M. Zhu, C. Dai, F. Wang, M. Tan, S. Hopyan, *et al.*, "Intracellular manipulation and measurement with multipole magnetic tweezers," *Science robotics*, vol. 4, no. 28, p. eaav6180, 2019.
- [3] C. Dai, Z. Zhang, Y. Lu, G. Shan, X. Wang, Q. Zhao, C. Ru, and Y. Sun, "Robotic manipulation of deformable cells for orientation control," *IEEE Transactions on Robotics*, vol. 36, no. 1, pp. 271–283, 2019.
- [4] L. Ren, N. Nama, J. M. McNeill, F. Soto, Z. Yan, W. Liu, W. Wang, F. Wang, and T. E. Mallouk, "3d steerable, acoustically powered microswimmers for single-particle manipulation," *Science advances*, vol. 5, no. 10, p. eaav3084, 2019.
- [5] Y. Ren, Q. Chen, M. He, X. Zhang, H. Qi, and Y. Yan, "Plasmonic optical tweezers for particle manipulation: principles, methods, and applications," *ACS nano*, vol. 15, no. 4, pp. 6105–6128, 2021.
- [6] Z. Zhang, J. Liu, X. Wang, Q. Zhao, C. Zhou, M. Tan, H. Pu, S. Xie, and Y. Sun, "Robotic pick-and-place of multiple embryos for vitrification," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 570–576, 2016.
- [7] S. Miao, J. Xu, Z. Jiang, J. Luo, X. Sun, X. Jiang, H. Wei, and Y.-H. Liu, "Microfluidics-enabled robotic system for embryo vitrification with real-time observation: Design, method, and evaluation," *IEEE/ASME Transactions on Mechatronics*, 2023.
- [8] D. Fan, F. Zhu, R. Cammarata, and C. Chien, "Electric tweezers," *Nano Today*, vol. 6, no. 4, pp. 339–354, 2011.
- [9] X. Li, C. C. Cheah, S. Hu, and D. Sun, "Dynamic trapping and manipulation of biological cells with optical tweezers," *Automatica*, vol. 49, no. 6, pp. 1614–1625, 2013.
- [10] M. P. Kummer, J. J. Abbott, B. E. Kratochvil, R. Borer, A. Sengul, and B. J. Nelson, "Octomag: An electromagnetic system for 5-dof wireless micromanipulation," *IEEE Transactions on Robotics*, vol. 26, no. 6, pp. 1006–1017, 2010.
- [11] T. Xu, C. Huang, Z. Lai, and X. Wu, "Independent control strategy of multiple magnetic flexible millirobots for position control and path following," *IEEE Transactions on Robotics*, vol. 38, no. 5, pp. 2875–2887, 2022.
- [12] L. Yang, Z. Yang, M. Zhang, J. Jiang, H. Yang, and L. Zhang, "Optimal parameter design and microrobotic navigation control of parallel-mobile-coil systems," *IEEE Transactions on Automation Science and Engineering*, 2022.
- [13] X. Du and J. Yu, "Image-integrated magnetic actuation systems for localization and remote actuation of medical miniature robots: A survey," *IEEE Transactions on Robotics*, vol. 39, no. 4, pp. 2549–2568, 2023.
- [14] Q. Wang, L. Yang, J. Yu, C.-I. Vong, P. W. Y. Chiu, and L. Zhang, "Magnetic navigation of a rotating colloidal swarm using ultrasound images," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 5380–5385.
- [15] J. Li, H. Wang, Q. Shi, Z. Zheng, J. Cui, T. Sun, P. Ferraro, Q. Huang, and T. Fukuda, "Biped walking of magnetic microrobot in oscillating field for indirect manipulation of non-magnetic objects," *IEEE Transactions on Nanotechnology*, vol. 19, pp. 21–24, 2019.
- [16] F. Ruggiero, V. Lippiello, and B. Siciliano, "Nonprehensile dynamic manipulation: A survey," *IEEE Robotics and Automation Letters*, vol. 3, no. 3, pp. 1711–1718, 2018.
- [17] J. Moura, T. Stouraitis, and S. Vijayakumar, "Non-prehensile planar manipulation via trajectory optimization with complementarity constraints," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 970–976.
- [18] Y. Jiang, Y. Jia, and X. Li, "Contact-aware non-prehensile manipulation for object retrieval in cluttered environments," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2023, pp. 10604–10611.
- [19] W. Chen, H. Zhou, B. Zhang, Q. Cao, B. Wang, and X. Ma, "Recent progress of micro/nanorobots for cell delivery and manipulation," *Advanced Functional Materials*, vol. 32, no. 18, p. 2110625, 2022.
- [20] S. Floyd, C. Pawashe, and M. Sitti, "Two-dimensional contact and noncontact micromanipulation in liquid using an untethered mobile magnetic microrobot," *IEEE Transactions on Robotics*, vol. 25, no. 6, pp. 1332–1342, 2009.
- [21] C. Pawashe, S. Floyd, E. Diller, and M. Sitti, "Two-dimensional autonomous microparticle manipulation strategies for magnetic microrobots in fluidic environments," *IEEE Transactions on Robotics*, vol. 28, no. 2, pp. 467–477, 2011.
- [22] R. Pieters, H.-W. Tung, S. Charreyron, D. F. Sargent, and B. J. Nelson, "Rodbot: A rolling microrobot for micromanipulation," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2015, pp. 4042–4047.
- [23] T. Petit, L. Zhang, K. E. Peyer, B. E. Kratochvil, and B. J. Nelson, "Selective trapping and manipulation of microscale objects using mobile microvortices," *Nano letters*, vol. 12, no. 1, pp. 156–160, 2012.
- [24] Z. Ye and M. Sitti, "Dynamic trapping and two-dimensional transport of swimming microorganisms using a rotating magnetic microrobot," *Lab on a Chip*, vol. 14, no. 13, pp. 2177–2182, 2014.
- [25] Z. Lin, X. Fan, M. Sun, C. Gao, Q. He, and H. Xie, "Magnetically actuated peanut colloid motors for cell manipulation and patterning," *ACS nano*, vol. 12, no. 3, pp. 2539–2545, 2018.
- [26] I. S. Khalil, A. Klingner, Y. Hamed, Y. S. Hassan, and S. Misra, "Controlled noncontact manipulation of nonmagnetic untethered microbeads orbiting two-tailed soft microrobot," *IEEE transactions on robotics*, vol. 36, no. 4, pp. 1320–1332, 2020.
- [27] Y. Dai, L. Jia, L. Wang, H. Sun, Y. Ji, C. Wang, L. Song, S. Liang, D. Chen, Y. Feng, *et al.*, "Magnetically actuated cell-robot system: Precise control, manipulation, and multimode conversion," *Small*, vol. 18, no. 15, p. 2105414, 2022.
- [28] I. Sinn, P. Kinnunen, S. N. Pei, R. Clarke, B. H. McNaughton, and R. Kopelman, "Magnetically uniform and tunable janus particles," *Applied Physics Letters*, vol. 98, no. 2, 2011.
- [29] T. Xu, Z. Hao, C. Huang, J. Yu, L. Zhang, and X. Wu, "Multimodal locomotion control of needle-like microrobots assembled by ferromagnetic nanoparticles," *IEEE/ASME Transactions on Mechatronics*, vol. 27, no. 6, pp. 4327–4338, 2022.
- [30] E. Bertrand, T. D. Blake, and J. De Coninck, "Influence of solid-liquid interactions on dynamic wetting: a molecular dynamics study," *Journal of Physics: Condensed Matter*, vol. 21, no. 46, p. 464124, 2009.
- [31] E. J. Dickinson, H. Ekström, and E. Fontes, "Comsol multiphysics@: Finite element software for electrochemical analysis. a mini-review," *Electrochemistry communications*, vol. 40, pp. 71–74, 2014.
- [32] H. Yu, T. Xie, S. Paszczynski, and B. M. Wilamowski, "Advantages of radial basis function networks for dynamic system design," *IEEE Transactions on Industrial Electronics*, vol. 58, no. 12, pp. 5438–5450, 2011.
- [33] M. Yu, K. Lv, H. Zhong, S. Song, and X. Li, "Global model learning for large deformation control of elastic deformable linear objects: An efficient and adaptive approach," *IEEE Transactions on Robotics*, vol. 39, no. 1, pp. 417–436, 2023.
- [34] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.
- [35] J.-J. E. Slotine, W. Li, *et al.*, *Applied nonlinear control*. Prentice hall Englewood Cliffs, NJ, 1991, vol. 199, no. 1.
- [36] H. Prautzsch, W. Boehm, and M. Paluszny, *Bézier and B-spline techniques*. Springer, 2002, vol. 6.
- [37] G. Bradski, "The opencv library," *Dr. Dobbs's Journal: Software Tools for the Professional Programmer*, vol. 25, no. 11, pp. 120–123, 2000.