Proceedings of The 2024 IEEE International
Conference on Real-time Computing and Robotics
June 24 - 28, 2024

2024 IEEE International Conference on Real-time Computing and Robotics (RCAR) | 979-8-3503-7260-1/24/$31.00 ©2024 IEEE | DOI: 10.1109/RCAR61438.2024.10670936

Dynamic Obstacle Avoidance for Magnetic Helical Microrobots Based on Deep Reinforcement Learning

Yukang Qiu, Yaozhen Hou*, Haotian Yang, Yigao Gao, Hen-Wei Huang, Qing Shi, Qiang Huang, *Fellow*, *IEEE*, and Huaping Wang

***Abstract*—Magnetic helical microrobots hold immense promise in biomedical domains owing to their compact size and efficient propulsion capabilities. However, navigating these microrobots through dynamic and unstructured environments, particularly when encountering numerous dynamic obstacles, remains a formidable challenge. In the study, a control framework based on deep reinforcement learning (DRL) with the objective of guiding a microrobot through dynamic obstacles towards specified target goals is introduced. Initially, we design and fabricate a microdrill capable of propulsion via external magnetic rotating fields produced by our magnetic actuation system. Subsequently, we construct a custom training environment, adhering to the OpenAI gym interface, to serve as the simulator for training purposes. Utilizing the proximal policy optimization algorithm, we conduct training of the navigation policy within this simulator. Simulations and experimental validations conducted in dynamic environments affirms the efficacy of the proposed method.**

I. INTRODUCTION

Magnetic helical microrobots have garnered significant interest in minimally invasive medicine owing to their distinctive ability to navigate through confined and enclosed spaces via remote manipulation[1-3]. Various approaches have been proposed to enhance microrobot navigation. Xu et al. proposed an image-based control method that successfully achieved planar path following and static obstacle avoidance[4]. J. Liu et al. developed a proxy-based sliding mode control strategy for 3D control of a helical microrobot, enabling efficient exploration of the shortest route within confined 3D spaces using existing path planning algorithms[5]. However, fully autonomous navigation in the complex and dynamic environment and dynamic obstacle avoidance for the microrobot remains challenging. When performing in vivo tasks, the microrobots would inevitably encounter moving obstacles, such as cell clusters and shed tissue. Therefore, it is necessary to endow the microrobot with the ability of dynamic obstacle avoidance to further promote its in vivo biomedical applications. Unlike navigation in a static environment, the microrobot cannot move along a predetermined path, on the contrary, it has to make optimal decisions in each step and adapt to the dynamic environment.

Efforts have been made to facilitate navigation of microrobots in dynamic environments by improving or combining traditional path planning algorithms. In a research by Q. Fan et al., the combination of static global path planning and dynamic local path re-planning was used to achieve static and dynamic obstacle avoidance in simulated vascular environments[6]. The process commenced with the generation of an optimized global path using the Improved Rapidly-exploring Random Trees (IRRT) algorithm[7], succeeded by the utilization of the Artificial Potential Field (APF) algorithm[8] to improve real-time performance during dynamic obstacle avoidance. Obstacle avoidance for a magnetic bead with up to 4 dynamic obstacles is demonstrated. However, the success rate of obstacle avoidance is strongly influenced by environmental parameters, and this method necessitates intricate adjustments to the parameters of two algorithms. Moreover, as the number of dynamic obstacles grows, frequent local path replanning could increase computational complexity, potentially diminishing real-time performance. Another study by T. Li et al. employed the fuzzy logic approach, based on human experience, to convert natural language control strategies into a lookup table[9]. The input variables, such as the location and distance of the obstacle to the microrobot, were characterized by concepts like "Left" or "Near," while the output was a heading angle change, such as "Forward" or "Back". However, the extension of fuzzy logic systems to accommodate more complex environments with numerous dynamic obstacles presents challenges, particularly concerning the management of intricate rule bases under such conditions.

The integration of deep reinforcement learning (DRL) into robotics, particularly in navigation, has introduced new possibilities. C. Wang et al. introduced a deep reinforcement learning (DRL) methodology for guiding unmanned aerial vehicles through vast and intricate environments[10]. M. Everett et al. showcased the efficacy of a collision avoidance algorithm based on deep reinforcement learning, GA3C-CADRL, in directing a fleet of four fully autonomous

This work was supported by National Key Research and Development Program of China under grant 2023YFB4705400, Postdoctoral Fellowship Program of CPSF under Grant BX20230459, Beijing Natural Science Foundation under grant 4232055, the National Natural Science Foundation of China under grant number 62073042 and 62222305. *(Corresponding author: Yaozhen Hou).*

Yukang Qiu, Yaozhen Hou, Haotian Yang and Yigao Gao are with the Intelligent Robotics Institute, School of Mechatronical Engineering, Beijing Institute of Technology, Beijing 100081, China. (e-mail: 3120210149@bit.edu.cn; 7520230117@bit.edu.cn; 1120230044@bit.edu.cn; 1120223543@bit.edu.cn).

Hen-Wei Huang is with the Laboratory for Translational Engineering, Harvard Medical School, Cambridge, MA 02139, USA (e-mail: hhuang27@bwh.harvard.edu).

Qing Shi and Qiang Huang are with the Beijing Advanced Innovation Center for Intelligent Robots and Systems, Beijing Institute of Technology, Beijing 100081, China (e-mail: shiqing@bit.edu.cn; qhuang@bit.edu.cn).

Huaping Wang is with the Key Laboratory of Biomimetic Robots and Systems, Beijing Institute of Technology, Ministry of Education, Beijing 100081, China (e-mail: wanghuaping@bit.edu.cn).

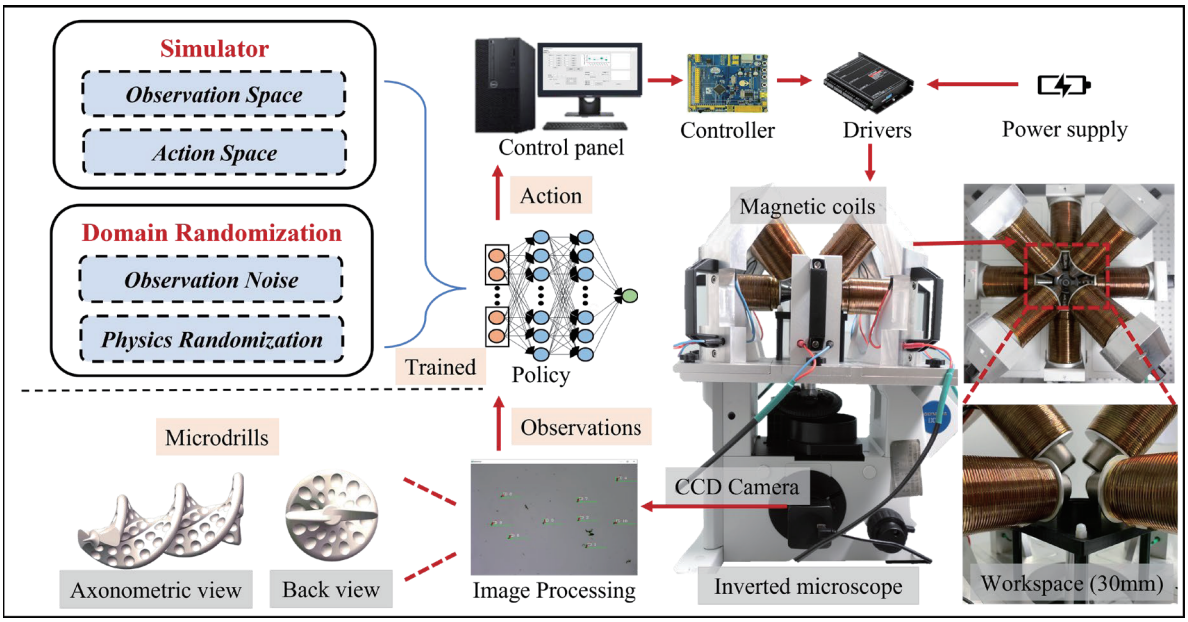

Fig. 1. Illustration of the whole system.

multirotors to avoid collision and navigating a ground robot at human walking speed among pedestrians[11]. The integration of microrobot control with DRL algorithms has the potential to enable optimal decision-making and adaptation to dynamic environments. However, due to the highly noisy, complicated, dynamic, and unstructured working environment of microrobots, it is challenging to accurately model such an environment in a virtual training setting. Consequently, the integration of DRL algorithms and microrobot control remains limited due to the challenge of constructing a virtual environment that faithfully replicate the dynamic and unstructured nature of real-world microrobot environment.

In the paper, we introduced a DRL-based control framework targeting goal-reaching and dynamic obstacle avoidance for magnetic helical microrobots. The primary contributions of this research are outlined below.

1) A DRL-based control framework is proposed and implemented for achieving dynamic obstacle avoidance and goal-reaching for a helical magnetic microrobot.
2) To enhance the efficiency of data collection for training, a custom training environment tailored to capture the crucial aspects of the navigation task for magnetic helical microrobots is developed.
3) A sim-to-real transfer method is incorporated into the training process for seamless transfer of the trained policy to real-world systems.

The paper is organized as follows: Section II provides an overview of the microdrill design and the magnetic actuation system. Section III discusses the details of the collision-free navigation model. Training and experimental results are presented in Section IV, followed by the conclusions in Section V.

II. SYSTEM SETUP

A helical magnetic microrobot is designed for manipulation. The microdrill measures 75 μm in length, with a pitch of 60 μm, wavenumber of 1.25, inner diameter of 4.2 μm, and cord radius of 12.3 μm. The microdrill was made of compounded biocompatible photoresist with a photoinitiator, which was fabricated by a high-precision 3D photolithography system (NanoScribe Photonic Professional GT). In addition, the microdrills were uniformly coated with magnetic nanoparticles for magnetic actuation. The manipulation of the microdrills is realized through the magnetic field generated by our magnetic actuation system. The magnetic actuation system comprises eight axial electromagnets enabling five-degree-of-freedom (5-DOF) motion control of the microdrill. Within a spherical workspace of 30 mm diameter, the system can generate a rotating magnetic field with a maximum intensity of 30 mT and a maximum gradient of 1.6 T/m. The detailed hardware specifications can be found in our prior publication[12]. When the microdrill is subjected to an external uniform magnetic field, the magnetic torque τ of the microdrill can be expressed as:

$$\tau = V\mathbf{M} \times \mathbf{B} \quad (1)$$

where V represents the volume of the microrobot, M represents its magnetization, and **B** denotes the external magnetic field. The torque tends to align the magnetic moment with the applied field[13]. By continuously rotating the applied field **B** in a circle on a two-dimensional plane, the microdrill undergoes continuous rotation around its helical axis to achieve propulsion. The swimming direction and forward speed of the microdrill can be controlled by manipulating the rotation direction and frequency of the magnetic field.

To facilitate the training of the DRL policy, a custom training environment both adhering to the OpenAI Gym interface[14] and abstracting the core physics of the navigation task is constructed as the simulator. To better transfer the policy

trained in simulation to the real-world systems, a domain randomization method is adopted during training. The trained policy receives real-world observations through real-time image capture and OpenCV image processing, then produce actions to navigate the microrobots .The overview of the whole system is illustrated in Fig.1.

III. COLLISION-FREE NAVIGATION MODEL BASED ON DRL

A. Customized Training Environment

1) Task: In our study, we aim to navigate a microdrill, driven by an external electromagnetic system, to a predetermined goal location while avoiding dynamic obstacles. The microdrills respond to a rotating magnetic field, with motion direction and speed regulated by the orientation and frequency of the external rotating magnetic fields, respectively. Within the simulation environment, each episode involves the random generation of a goal location within the environment's boundaries, alongside a set number of obstacles randomly placed along either the bottom or rightmost edge of the environment. The position of one generated obstacle i can be expressed as follows:

$$\mathbf{p_i} = \begin{cases} x \sim U[d, W],\ y = 0 & \text{if } r = -1 \\ x = W,\ y \sim U[0, H-d] & \text{if } r = 1 \end{cases} \tag{2}$$

where $U[a,b]$ denotes a uniform distribution between a and b, d is the safe distance away from the boundary, r is a random variable drawn from a discrete uniform distribution $\{-1,1\}$, and W and H are the width and height of the environment, respectively.

The direction of each obstacle's movement varies in each timestep, but generally, they move toward the upper left corner of the environment. The speed of an obstacle i in each timestep is:

$$\mathbf{v_i} = \begin{pmatrix} v_{ix} \\ v_{iy} \end{pmatrix} = \begin{pmatrix} U[-v_{\max}, v_{\min}] \\ U[v_{\min}, v_{\max}] \end{pmatrix}, \quad i = 1, 2, \ldots, n \tag{3}$$

During training, the environment maintains a constant number of obstacles, replenishing those that move beyond the boundaries with newly generated obstacles based on Equation (2). The microdrill's task is considered a failure if it collides with either an obstacle or the environment's boundaries, while successful navigation is achieved upon reaching the designated goal.

2) Observation Space: The observation space is an 8-dimensional box as shown in TABLE I.

TABLE I. OBSERVATION SPACE

Parameters	Meanings
p_x	the x-coordinate of the microdrill's current position
p_y	the y-coordinate of the microdrill's current position
θ	the orientation (in radians) of the microdrill's heading
r	the radius of the circumscribed circle of the microdrill
g_x	the x-coordinate of the goal's position
g_y	the y-coordinate of the goal's position
t	the current timesteps in the episode

3) Action Space: In this environment, we focus on the key parameters that govern the motion of a helical micro swimmer, specifically, the direction of the rotating magnetic field. The action space A is a 1-dimensional box with lower and upper bounds of -1 and 1, respectively. This action is used to adjust the orientation of the microdrill, which is updated by adding the product of the action and $\pi/6$ to the current orientation. In this way, we limit the change in orientation of the microdrill to one timestep within the range of $[-\pi/6,\ \pi/6]$.

$$A = [\theta_d] \tag{4}$$

4) Reward Function: The reward function r serves as a pivotal component in our system, evaluating a reward signal composed of four distinct terms: navigation bonus b_n, obstacle penalty p_o, time penalty p_t, and velocity potential p_v. The navigation bonus, governed by an attractive potential, incentivizes the microdrill to progress towards the goal by providing a reward inversely proportional to its distance from the goal position. As the microdrill approaches the goal, the reward magnitude increases, fostering efficient navigation. Conversely, the obstacle penalty discourages the microdrill from proximity to obstacles, imposing penalties relative to the inverse of obstacle distances. The penalty intensifies as the microdrill draws nearer to obstacles, determined by a repulsive coefficient that governs the strength of the repulsive force. The time penalty discourages prolonged navigation periods, imposing a negative reward proportional to the time taken by the microdrill to reach its destination. This constant penalty serves to promote timely navigation. Additionally, the velocity potential reinforces obstacle avoidance behavior, penalizing the microdrill for movements towards obstacles and encouraging avoidance. This penalty is determined by the dot product of relative velocity and obstacle unit vectors, with positive values indicating concordant movement and negative values signaling movement towards obstacles. The velocity potential parameter regulates the strength of the penalty, ensuring appropriate responses to obstacle proximity.

TABLE II. REWARD FUNCTION

1	Calculate navigation bonus based on the attraction coefficient and distance to the goal: $b_n = c_a \cdot \frac{1}{d_{2g}}$
2	Calculate obstacle penalty p_o based on the repulsive coefficient c_r, distance between the microdrill and the ith obstacle d_{2i} and the safe distance d_{safe}: $p_o = -c_r \cdot \sum_i^n \left(\frac{1}{d_{2i}} - \frac{1}{d_{safe}} \right)$
3	Calculate time penalty p_t based on a constant penalty k_t: $p_t = -k_t$

Calculate velocity potential p_v by summing the penalties for each obstacle:
For each obstacle i **do**
  Calculate relative velocity between the microdrill and the i th obstacle $\vec{v}_{rel}$
  Get the unit vector pointing from the i th obstacle to the microdrill $\vec{a}_i$
  if $\vec{v}_{rel} \cdot \vec{a}_i > 0$ **then**
    $p_v += 0$
  else
    $p_v += k_v \cdot \vec{v}_{rel} \cdot \vec{a}$
  end if
End for

Calculate total reward r : $r = b_n + p_o + p_t + p_v$
Return total reward

IV. EXPERIMENTS

A. Training

We build a custom training environment as shown in Fig. 2 (a). Due to the highly complex and noisy environment in micro scale, completely modelling the physics of such environment in the simulator is challenging. Instead, researchers often opt to abstract certain key aspects of the real world and customize specific simulators addressing specific requirements and constraints for simplicity[15-17]. To simplify the simulation environment, we adopt the following assumptions:

1. The speed of the microdrill is regulated by the frequency of the rotating magnetic field. Given that the microrobot typically operates at a fixed frequency lower than the step-out frequency, we maintain a constant speed in our simulation.
2. The direction of the microdrill's motion is determined by the direction of the rotating magnetic field. Hence, during navigation, our primary parameter of control is the rotating direction of the field.
3. To avoid collision, the microdrill and dynamic obstacles must maintain a safe distance. In our simulation, we assess potential collisions by comparing the distance between the microdrill and an obstacle to the sum of the radii of their circumscribed circles.

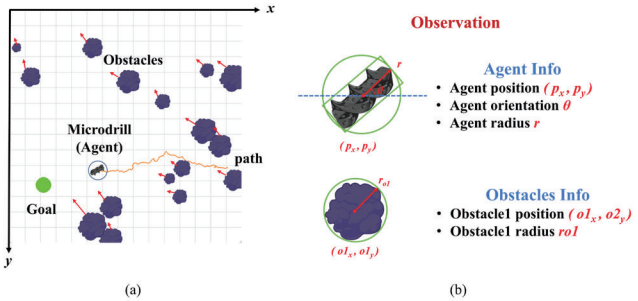

Fig. 2. (a) Schematic of the custom simulator. (b) Illustration of the observation of the microdrill and obstacles.

Given the continuous nature of both the action space and observation space in our custom environment, we have chosen proximal policy optimization (PPO) as our training algorithm. PPO stands out for its sample efficiency, utilizing a clipped surrogate objective to mitigate the risk of overly drastic policy updates and thus preventing divergence issues[18]. With its stochastic policy, PPO inherently balances the exploration-exploitation trade-off, aiding the microdrill in discerning optimal policies within the complexities of our dynamic environment. Our training process, outlined in TABLE III, begins with the initialization of an "ObstacleAvoidanceEnv" environment, purposefully crafted for microdrill navigation towards a goal while avoiding obstacles. To enhance sample efficiency during training, we generate a vectorized environment with multiple parallel instances of the obstacle avoidance environment. To gap the difference between reality and simulation, we randomize several key parameters in the environments during initialization. PPO serves as our chosen reinforcement learning algorithm, with the policy function employing a multilayer perceptron (MLP), a neural network that takes the environment state as input and outputs the probability distribution over actions. Additionally, an evaluation callback is configured to periodically assess the model's performance on the environment, logging relevant metrics and facilitating early stopping if performance criteria are met. Throughout training, the policy guides the microdrill's actions at each timestep, with PPO employing a combination of policy and value iteration for iterative policy improvement. The utilization of a clipped surrogate objective ensures stable policy updates by constraining the objective function to prevent large updates that might compromise training stability. Upon completion of training, the PPO model is saved for subsequent evaluation, testing, and deployment.

TABLE III. TRAINING PROCESS

Training Process

Initialization:
1 Initialize environment:
  env←*make_vec_env(ObstacleAvoidanceEnv, n_envs)*
2 Initialize model:
  model←*PPO('MlpPolicy', env, verbose, tensorboard_log, learning_rate, batch_size)*
3 Initialize evaluation callback:
  eval_callback←*EvalCallback(env, log_path=, eval_freq=total_timesteps // 10)*

Training:
4 **For** iteration=1, 2…, total_timesteps **do**:
5 Collect data: *trajectories*←*CollectData(env, model)*
6 **For** epoch=1, 2…, K **do**
7 1 Compute policy probabilities and values: *policy_probs, values*←*EvaluatePolicy(model, trajectories)*
8 2 Compute advantages and returns:

advantages, returns ← *ComputeAdvantagesReturns(trajectories, values)*

Optimize policy and value function: *OptimizePolicyValue(model, trajectories, advantages, policy_probs)*

Train model: *model.learn(total_timesteps, eval_callback)*

Save model:

Model.save(save_path)

Training involved 500k total time steps on an NVIDIA GeForce RTX 2060 with 22 GB of memory, completing in 19 minutes and 11 seconds. The training metrics, shown in Fig. 3, indicate that the "episode mean length" decreases steadily after an initial exploration phase, while the "episode mean reward " steadily increases until peaking at approximately 200k time steps. This trend demonstrates the microdrill's progressive refinement of its strategy for more efficient goal achievement, reaching optimal performance around 200k time steps.

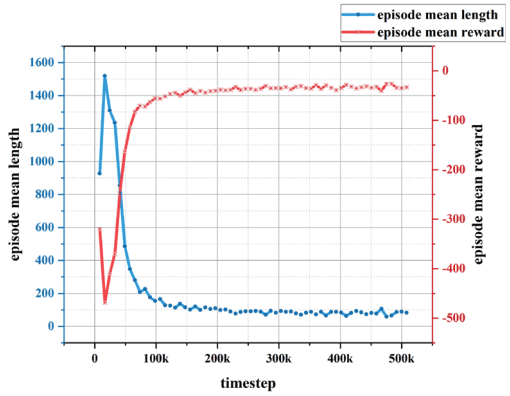

Fig. 3. The average length of episodes and the average reward of episodes during training.

B. Testing in simulation

Simulation tests were conducted to evaluate the obstacle avoidance efficacy of the trained policy. The microdrill and goal were randomly positioned within an environment containing 12 dynamic obstacles, all obstacles tended to move toward the upper left corner. Failure criteria included boundary exceedance, obstacle collision, or time limit surpassing. Success rate, indicating the microdrill's ability to reach its goal without colliding with obstacles, was computed over a thousand simulation test episodes. The result showed the success rate reaches 93.1% with a mean reward of -52.1, indicating efficacy for dynamic obstacle avoidance.

C. Experiment on a real-world system

The magnetic actuation system for experiment is illustrated in Fig. 1 . The microdrill, the obstacles, and the goal, are initially placed at random positions within the environment. The yellow circle surrounding the microdrill denote its circumcircle. Collision with obstacles or goal attainment is determined by assessing if the distance between two objects is smaller than the sum of their circumradii. The green track indicate the path of the microdrill. Tasked with pursuing the goal while evading obstacles, the microdrill adjusts its course accordingly. Upon reaching a goal, a new goal is randomly generated for the microdrill. In real-world experiments, we utilize image processing to simulate artificial obstacles, randomly placing them with varying radii within a predefined range. The number of obstacles within boundaries is fixed at 12. Fig. 4 illustrates the obstacle generation process, where obstacles originate from the right and bottom edges, flowing towards the upper left. Fig. 4(a) demonstrates the microdrill's avoidance maneuver when an obstacle approaches. In Fig. 4(b), when surrounded by two dynamic obstacles, the microdrill prioritizes avoiding the closest one, resulting in a "bouncing" pattern through the obstacles. Fig. 4(c) depicts the microdrill's behavior when an obstacle is near the goal, prioritizing obstacle avoidance before circling back to the goal. Finally, Fig. 4(d) displays the complete navigation path, with the red flags indicating reached goals. The experiment indicates that the trained policy can be successfully transfer to the real-world systems and can navigate the microdrill in complex and dynamic environments.

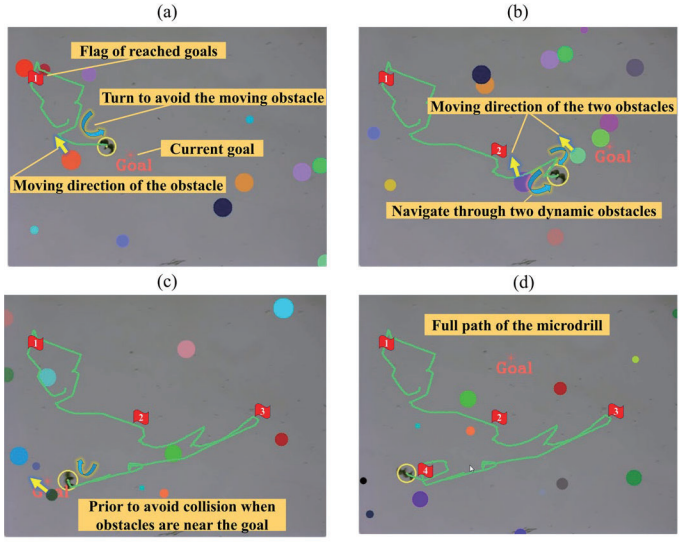

Fig. 4. Image sequence capturing the microdrill chasing a sequence of goals while avoiding dynamic obstacles.
view.

V. Conclusion

This paper presents a Deep Reinforcement Learning (DRL)-based control method of goal-reaching and dynamic obstacle avoidance for a microdrill. The control strategy integrates custom training environments, DRL algorithms, a magnetic actuation system, and real-time visual tracking methods. Initially, we fabricate a helical drill-like microrobot actuated by a rotating magnetic field. To gather data effectively, we construct a custom DRL training environment compliant with the OpenAI Gym interface, abstracting the core physics of the navigation task. Leveraging the PPO method tailored to our environment's characteristics, we train the policy and incorporate sim-to-real transfer techniques to enhance adaptability in real-world scenarios. Simulation and experimental results demonstrate the successful accomplishment of goal-reaching and dynamic obstacle

avoidance tasks with notable adaptability, showcasing potential applications in biomedical in-vivo settings. In the future, efforts will be devoted to improve the adaptability of the method so that it can be applied to a wider range of scenarios.

REFERENCE

[1] 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.

[2] M. Sitti, "Miniature soft robots — road to the clinic," *Nat Rev Mater*, vol. 3, no. 6, Art. no. 6, Jun. 2018.

[3] Y. Dong, L. Wang, V. Iacovacci, X. Wang, L. Zhang, and B. J. Nelson, "Magnetic helical micro-/nanomachines: Recent progress and perspective," *Matter*, vol. 5, no. 1, pp. 77–109, Jan. 2022.

[4] T. Xu, Y. Guan, J. Liu, and X. Wu, "Image-Based Visual Servoing of Helical Microswimmers for Planar Path Following," *IEEE Transactions on Automation Science and Engineering*, vol. 17, no. 1, pp. 325–333, Jan. 2020.

[5] J. Liu et al., "3-D Autonomous Manipulation System of Helical Microswimmers With Online Compensation Update," *IEEE Transactions on Automation Science and Engineering*, vol. 18, no. 3, pp. 1380–1391, Jul. 2021.

[6] Q. Fan, G. Cui, Z. Zhao, and J. Shen, "Obstacle Avoidance for Microrobots in Simulated Vascular Environment Based on Combined Path Planning," *IEEE Robotics and Automation Letters*, vol. 7, no. 4, pp. 9794–9801, 2022.

[7] L. S, "Rapidly-exploring random trees : a new tool for path planning," *Research Report 9811*, 1998.

[8] P. Vadakkepat, K. C. Tan, and W. Ming-Liang, "Evolutionary artificial potential fields and their application in real time robot path planning," *in Proceedings of the 2000 Congress on Evolutionary Computation. CEC00 (Cat. No.00TH8512)*, Jul. 2000, pp. 256–263 vol.1.

[9] T. Li et al., "Autonomous Collision-Free Navigation of Microvehicles in Complex and Dynamically Changing Environments," *ACS Nano*, vol. 11, no. 9, pp. 9268–9275, Sep. 2017.

[10] C. Wang, J. Wang, Y. Shen, and X. Zhang, "Autonomous Navigation of UAVs in Large-Scale Complex Environments: A Deep Reinforcement Learning Approach," *IEEE Transactions on Vehicular Technology*, vol. 68, no. 3, pp. 2124–2136, Mar. 2019.

[11] M. Everett, Y. F. Chen, and J. P. How, "Collision Avoidance in Pedestrian-Rich Environments with Deep Reinforcement Learning," *IEEE Access*, vol. 9, pp. 10357–10377, 2021.

[12] Y. Hou et al., "Design and Control of a Surface-Dimple-Optimized Helical Microdrill for Motions in High-Viscosity Fluids," IEEE/ASME Transactions on Mechatronics, pp. 1–11, 2022.

[13] 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 Trans. Robot.*, vol. 26, no. 6, pp. 1006–1017, Dec. 2010.

[14] G. Brockman et al., "OpenAI Gym." *arXiv*, Jun. 05, 2016.

[15] K. Arulkumaran, M. P. Deisenroth, M. Brundage, and A. A. Bharath, "Deep Reinforcement Learning: A Brief Survey," *IEEE Signal Process. Mag.*, vol. 34, no. 6, pp. 26–38, Nov. 2017.

[16] M. Cai et al., "Deep Reinforcement Learning Framework-Based Flow Rate Rejection Control of Soft Magnetic Miniature Robots," *IEEE Transactions on Cybernetics*, pp. 1–13, 2022.

[17] M. Witman, D. Gidon, D. B. Graves, B. Smit, and A. Mesbah, "Sim-to-real transfer reinforcement learning for control of thermal effects of an atmospheric pressure plasma jet," *Plasma Sources Sci. Technol.*, vol. 28, no. 9, p. 095019, Sep. 2019.

[18] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal Policy Optimization Algorithms." *arXiv*, Aug. 28, 2017.