

Steering Micro-Robotic Swarm by Dynamic Actuating Fields

Qianwen Chao^{1,2}, Jiangfan Yu¹, Chengkai Dai¹, Tiantian Xu¹, Li Zhang¹,
Charlie C.L. Wang^{1,3†}, and Xiaogang Jin²

Abstract—We present a general solution for steering micro-robotic swarm by dynamic actuating fields. In our approach, the motion of micro-robots is controlled by changing the actuating direction of a field applied to them. The time-series sequence of actuating field’s directions can be computed automatically. Given a target position in the domain of swarm, a governing field is first constructed to provide optimal moving directions at every points. Following these directions, a robot can be driven to the target efficiently. However, when working with a crowd of micro-robots, the optimal moving directions on different agents can contradict with each other. To overcome this difficulty, we develop a novel steering algorithm to compute a statistically optimal actuating direction at each time frame. Following a sequence of these actuating directions, a crowd of micro-robots can be transported to the target region effectively. Our steering strategy of swarm has been verified on a platform that generates magnetic fields with unique actuating directions. Experimental tests taken on aggregated magnetic micro-particles are quite encouraging.

I. INTRODUCTION

Remotely actuated micro-swimmers are very promising to act as micro-robots in a variety applications (e.g., the applications of micro-manipulation reported in [1]). Artificial bacterial flagella is an efficient and effective tool for micro-manipulation, with which the motion of micro-particles – can be controlled (ref. [2]). A cell holder fabricated in micro-scale was reported in [3] to capture and transport a living cell. Natural micro-manipulator like magnetotactic bacteria were also used to deliver objects in micro-scale. Martel [4] aggregates magnetotactic bacteria in a region of $80 \mu\text{m} \times 20 \mu\text{m}$ and transports them as a single particle. This method is hard to be applied to move a crowd of distributed micro-robots. How to transport a large number of micro-robots simultaneously is still challenging.

In literature, multiple stimuli such as magnetic field [5], chemical fuel [6], acoustic wave [7], light [8], etc. have caught more and more attention to be used in the multi-agent motion control of micro-robots. Different path planning and control algorithms have been developed. Kim et al. [9] use realtime feedback control and the *Rapidly-exploring Random Tree* (RRT) for path planning to control the movement of magnetotactic *T. pyriformis* as micro-robots. Ou et al. [10] investigate the motion control of *T. pyriformis* cells using the *Model Predictive Control* (MPC) algorithm. These control

*This work was supported by HKSAR RGC GRF CUHK/417812 and CUHK/14209514 and HKSAR ITF ITS/065/14.

¹Department of Mechanical and Automation Engineering, Chinese University of Hong Kong.

²State Key Lab of CAD&CG, Zhejiang University

³Department of Design Engineering, Delft University of Technology.

[†]Corresponding Author (Email: c.c.wang@tudelft.nl)

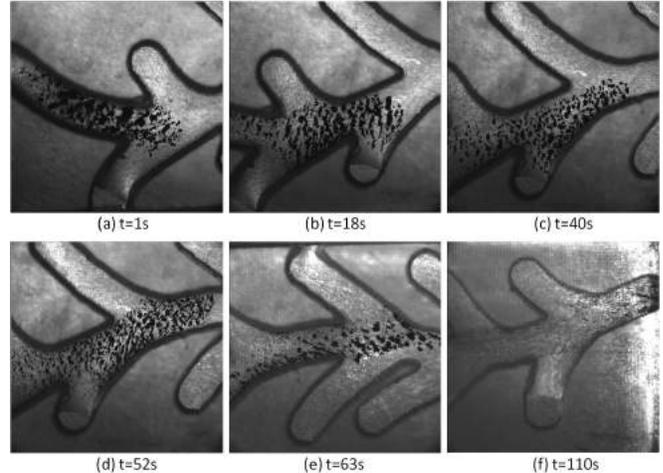


Fig. 1. Optimal steering of distributed micro-robots by our approach in a channel using dynamic magnetic fields. Micro-robots employed in this experiment are aggregated from nano-particles.

strategies mainly focus on single agent. To work on more agents, Ou et al. [11] extended the basic MPC algorithm with particle swarm optimization to control multiple robots simultaneously moving to their targets while avoiding obstacles. However, the number of micro-robots in their experiments is less than five. It is not clear if their method has a good scalability to be extended to handle hundreds of micro-robots as shown in Fig.1. Moreover, most of the existing methods are tested in simple environments. A general method for steering micro-robotic swarm needs to be investigated.

In this paper, a method is introduced for steering micro-robotic swarm using dynamic actuating fields. We make three contributions:

- Distributed crowds with many micro-robots can be well steered by the sequence of actuating field’s directions generated by our algorithm.
- The method can automatically generate sequence of actuating directions in environments with complex shape/topology and different obstacles.
- Our approach is general and can be applied to different types of micro-robots driven by different actuating fields.

Benefits of this method have been verified in experimental tests at the end of this paper.

The rest of this paper is structured as follows. Section II presents the general methodology of our approach. Governing field is generated in Section III to provide optimal moving directions at every points in the domain of interest.

Our steering algorithm is presented in Section IV, the validity of which is verified by the experimental tests described in Section V. Lastly, our paper ends with conclusion and discussion in Section VI.

II. METHODOLOGY

The methodology of our approach for steering micro-robotic swarm is presented in this section. The benefit and challenges of field-based steering is first discussed. We then brief the strategy of our steering method using dynamic actuating fields.

A. Field-based Steering

In prior work of micro-manipulation, micro-robots are mainly transported in a sequential execution manner (e.g., [1], [12]). Although it is effective, efficiency could become a challenge when the number of agents increases. To overcome this difficulty, field-based steering catches more and more attention as such a method can transport micro-robots in a parallel way – a large number of agents are moved simultaneously.

We investigate the method of using dynamic actuating fields for swarm steering in this paper. Here we assume all forces applied to micro-robots by a field are along the same direction at any particular time, which is easy to be realized by an existing hardware setup (e.g., the electromagnetic device shown in Fig.2). The directions of forces at different times can be numerically controlled by a computer system. We also assume that the field strength is high enough to ignore each micro-robot’s individual behavior, such as mobility and possible collisions between each other.

Given such a hardware setup for generating dynamic actuating fields, we need to compute a time-series sequence of field’s directions so that the micro-robots can be transported from their current positions to a target region. This is not easy when dealing with a large number of agents. First, we need a method that can efficiently assign optimal moving directions to every agents distributed in a large area. Moreover, the moving directions of different agents could contradict with each other. New steering strategy and algorithm must be investigated to solve these problems.

B. Steering Strategy

Like flocks of birds and ant colonies [13], micro-robots such as aggregated nano-particles, cells, bacteria, etc. can exhibit complex swarm behaviors when large numbers of simple agents react to local information. We treat the steering problem of multi-robotic swarm as a self-organized system. Specifically, we propose a general solution to transport them to the target region in arbitrary shaped environments, no matter in channels or with obstacles. Each micro-robot in the swarm can be regarded as an agent in the crowd. Given the position of a target \mathbf{q}_t in an environment, we compute a smooth goal-directed vector field to represent the expected moving direction for every points in the domain of interest, Ω . First, the cost of a point on the boundary of Ω is assigned as the in-domain distance to \mathbf{q}_t , which can be

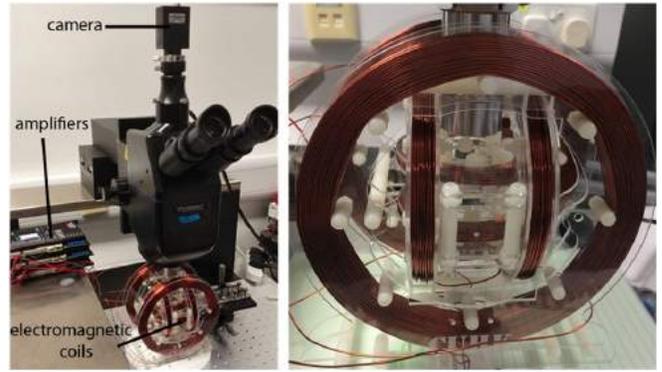


Fig. 2. The electromagnetic device used in our experiments consists of a three-axis Helmholtz electromagnetic coils, a microscope, a CCD camera and amplifiers. Dynamic magnetic fields can be generated by this setup, and the direction of field’s actuating is numerically controlled by rotating the coils.

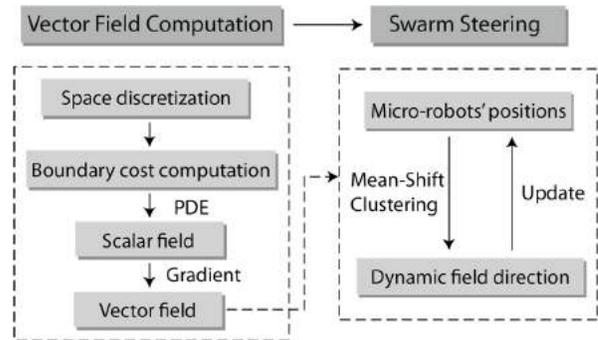


Fig. 3. A schematic overview of our approach for steering multi-robotic swarm by dynamic actuating fields.

obtained by the A^* algorithm after tessellating the domain into a discrete representation (e.g., regular grids employed in our work). Taking these costs at boundary points as the Dirichlet boundary conditions, a smooth scale field, $\phi(\mathbf{p})$ ($\forall \mathbf{p} \in \Omega$), can then be obtained by solving the Poisson’s equation, which results in a Harmonic field. Gradients of $\phi(\mathbf{p})$, $\mathbf{g}(\mathbf{p}) = \nabla\phi(\mathbf{p})$, are employed as the vector field for governing the movement of agents in our approach.

Caused by the reason that distributed micro-robots in the domain may have different ‘optimal’ moving directions on each individual, it is challenging to determine an ‘optimal’ common force direction to be applied to all agents. When a naïve average is employed, the movement of swarm can be ‘stuck’ due to the contradiction. As a result, agents cannot reach their target position effectively. We introduce a mean-shift based method to compute an optimal force direction satisfying the majority of demands. In addition, to ensure all micro-robots keep moving in a collective way (i.e., to prevent sticking micro-robots to the boundary of Ω), special rules for environmental collision response have been incorporated into the decision strategy of field directions.

In summary, we develop a new technique for steering multi-robotic swarm by dynamic actuating field, which can only generate forces in a unique direction at each time frame. This is a parallel execution so that the multi-robots are

simultaneously transported from their initial positions to a common target all together. Figure 3 provides a schematic overview of our approach.

III. COMPUTATION OF GOVERNING FIELD

Given an environment description with static obstacles and a specified position \mathbf{q}_t as target, a scalar field, $\phi(\mathbf{p})$, is defined over the domain of motion, Ω . $\forall \mathbf{p} \in \Omega$, $\phi(\mathbf{p})$ presents a cost with respect to the length of a path moving from \mathbf{p} to \mathbf{q}_t . There are usually two demands on the paths of robotic swarm (ref. [14]): 1) fast (short) and 2) natural (smooth). To meet these demands, we construct $\phi(\mathbf{p})$ according to the intrinsic-distances to \mathbf{q}_t and meanwhile enforcing its smoothness. Specifically, $\phi(\mathbf{p})$ can be defined as a Harmonic function which satisfies the Poisson's equation as

$$\nabla^2 \phi = 0 \quad (1)$$

where ∇^2 is the Laplacian operator. This partial differential equation can be solved by applying certain boundary conditions. The resultant field, $\phi(\mathbf{p})$, has C^2 -continuity in the whole domain Ω . To achieve a higher order of continuity, the second order of Laplacian is usually employed [15], that is.

$$\nabla^2(\nabla^2 \phi) = \nabla^4 \phi = 0 \quad (2)$$

Solving the above equation needs a discretization of Ω , and the boundary conditions must also be enforced to avoid obtaining a trivial solution. For motions of micro-robots in our experiments, the free space of robots in the environment can be indicated by discrete domains. Two-dimensional regular grids are employed in our implementation. The benefit is two-fold. First, regular grids are easy to be constructed from binary images. This can be directly applied to many biomedical applications, the input of which is biomedical images. Second, the formulas for computing Laplacian on regular grids are well defined and will not suffer from the problems of Laplacian on irregular meshes (ref. [16]). As a result, with the help of Laplacian operator \mathbf{L} defined on all grid nodes in Ω , the field values \mathbf{x} can be determined by

$$\mathbf{L}^2 \mathbf{x} = 0. \quad (3)$$

The boundary conditions imposed in our framework make $\phi(\mathbf{q}_t) = 0$ and give maximal value on the boundary of Ω (denoted by $\partial\Omega$). One simple solution is

$$\phi(\mathbf{p}) = 1 \quad (\forall \mathbf{p} \in \partial\Omega). \quad (4)$$

Numerical system of Eq.(3) by imposing these constraints can be determined by removing their corresponding rows from the linear system and substituting the known variables into other rows. The field values on all grid nodes can then be determined by solving the linear system processed in this way. Although this results in a smooth scalar field for $\phi(\cdot)$, the field values do not reflect the intrinsic-distances to the target in Ω . Moreover, the gradients of field, which will be used as a governing field of swarm, tend to push multi-robots towards the media-axis of Ω (see Fig.4(a) for an example). Since our task is to transport micro-robots to their targets as

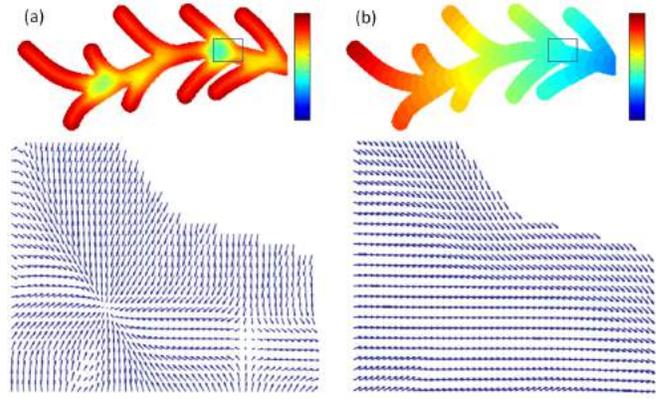


Fig. 4. The comparison of fields between (a) imposing a common boundary value on $\phi(\mathbf{p})$ and (b) using intrinsic-distance based field values for boundary points.

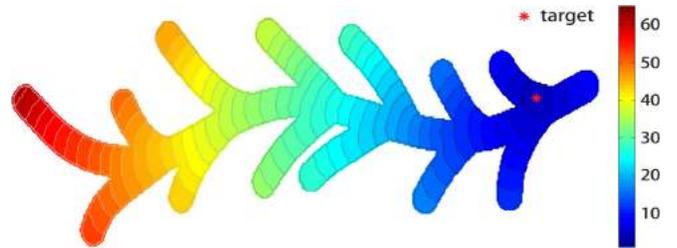


Fig. 5. Colormap of the scalar field $\phi(\mathbf{p})$ for a dendritic channel, where isocurves are also provided to illustrate the trend of ϕ 's gradients.

soon as possible, we wish the cost on each point \mathbf{p} is equal to the length of its shortest path to the target in the domain (i.e., the intrinsic-distance $\delta(\mathbf{p}, \mathbf{q}_t)$ and the path must completely lie in Ω). The boundary condition can then be imposed as

$$\phi(\mathbf{p}) = \delta(\mathbf{p}, \mathbf{q}_t) \quad (\forall \mathbf{p} \in \partial\Omega). \quad (5)$$

The intrinsic-distance $\delta(\mathbf{p}, \mathbf{q}_t)$ can be efficiently computed by using A^* algorithm [17]. As can be found in Fig.4(b), the governing field resulted from these boundary conditions offers a very good guidance of flowing directions, which are similar to laminar flow near the boundary of Ω . This is very similar to the collective behavior of swarm what we observe from nature.

Again, in a discrete representation of Ω , values of $\phi(\cdot)$ are determined on the grid nodes by solving Eq.(3). For an arbitrary point $\mathbf{p} \in \Omega$, its field value can be obtained by a local parameterization proportional to the inverse distances to its k -nearest neighboring grid nodes \mathbf{p}_j .

$$\mathbf{f}(\mathbf{p}) = \sum_{j=1}^k w_j \mathbf{f}(\mathbf{p}_j) \quad (6)$$

$$w_j = \frac{\|\mathbf{p} - \mathbf{p}_j\|^{-1}}{\sum_{i=1}^k \|\mathbf{p} - \mathbf{p}_i\|^{-1}},$$

In our implementation, $k = 24$ is used. By this method, the field value of any point in the domain Ω is well defined.

Isocurves generated by the field-values indicate that moving along the gradient direction of $\phi(\mathbf{p})$ can promptly reach

the target from any region in Ω (see Fig.5). Therefore, the governing field $\mathbf{f}(\mathbf{p})$ is defined as the normalized gradient of $\phi(\mathbf{p})$

$$\mathbf{f}(\mathbf{p}) = \frac{\nabla\phi(\mathbf{p})}{\|\nabla\phi(\mathbf{p})\|}. \quad (7)$$

The gradients are easy to be evaluated on regular grid nodes by numerical differentiations. For each grid, its field value $\mathbf{f}(\mathbf{p})$ is assigned as the average of values on its four corner nodes. After determining the governing field $\mathbf{f}(\mathbf{p})$, the vectors are referred as ‘optimal’ moving directions to supervise the movement of micro-robots.

IV. STEERING ALGORITHM

The governing field, $\mathbf{f}(\mathbf{p})$, computed above can be regarded as the expected moving directions at every points for the micro-robotic swarm. Micro-robots located at different places will have different ‘optimal’ moving directions. However, the currently available hardware platform can only generate a single actuating direction at each time frame. Simply averaging the directions of all agents will easily make the swarm stuck at a place when parts of agents are willing to move in opposite directions. Considering that the task of our work is to steer micro-robotic swarm to a target efficiently, we introduce the mean-shift clustering method [18], [19] to compute an optimal field direction that can statistically satisfy the majority of a crowd and therefore avoid being stuck. Here, optimum stands for a direction that swarm can reach the target with a maximal speed.

A. Mean-shift and Feature Vector

Mean-shift is a non-parametric feature space analysis technique for locating the maxima of a density function, which has been widely used in computer vision and image processing [20]. Considering a set of points in a k -dimensional feature-space, a window function is selected as a kernel. The mean-shift algorithm iteratively shifts this kernel to a higher density region until convergence. Every shift is defined by a mean-shift vector pointing toward the maximal increasing direction in terms of the density. Here, the kernel is shifted to the centroid (or the mean) of all points falling in the local support of the window function, where the centroid is called the *mean* point. Different types of kernel functions may lead to different methods for evaluating the mean.

In our problem, the feature space is spanned by the moving direction \mathbf{v} (unit vector) of each micro-robot – i.e., a 2D vector for a planar transportation problem. The initial feature vector of an agent a is defined by the governing field as $\mathbf{v}_a = \mathbf{f}(\mathbf{p}_a)$. The common steering direction at a time frame is determined later by the mean-shift of feature vectors of all agents in the swarm \mathcal{S} . See Fig.6 for an illustration, where the green dots represent the feature vectors and the red crosses denotes the mean point that is updated during the iterations of mean-shift.

B. Steering with Mean-shift

The window function controls the size of population considered in each iteration of mean-shift. Too large or

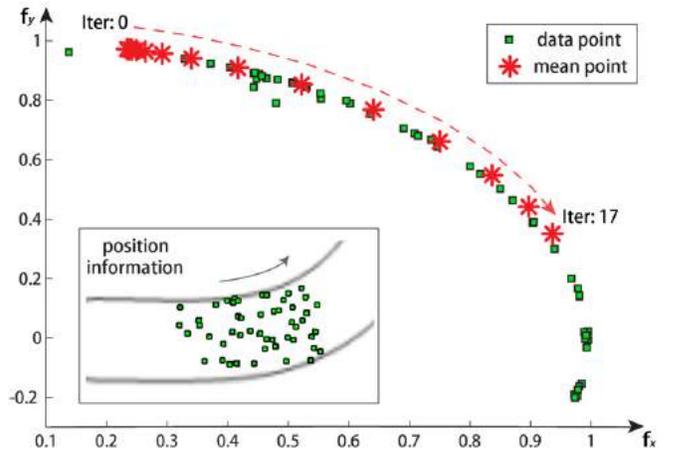


Fig. 6. An illustration of the mean-shifting process applied on feature vectors (green dots) of all agents to determine the ‘optimal’ steering direction to be generated by the dynamic actuating field. The red crosses represent the *mean* points computed during iterations.

too small will not provide a statistically meaningful result. As our feature vector is evaluated in the angle space, we set the window size as $W = 60^\circ$, which implies the maximally allowed difference with the mean. Specifically, when $\theta(\mathbf{v}_a, \mathbf{v}_b)$ returns the angle between two vectors \mathbf{v}_a and \mathbf{v}_b , the window function is defined as

$$\varpi(\mathbf{v}, \bar{\mathbf{v}}) = \begin{cases} 1, & \theta(\mathbf{v}, \bar{\mathbf{v}}) \leq W \\ 0, & \theta(\mathbf{v}, \bar{\mathbf{v}}) > W \end{cases} \quad (8)$$

where $\bar{\mathbf{v}}$ is the mean point.

The mean-shift algorithm to determine an optimal steering direction consists of the following steps:

- Step 1): At the very beginning of our steering algorithm, we randomly pick a point from the swarm \mathcal{S} to serve as the initial guess of the mean point $\bar{\mathbf{v}}$. After the first step of steering, position of the mean point in previous time-step can be used as the initialize value.
- Step 2): For each agent i within the window of the kernel, we compute its weight for re-estimating the mean using a kernel function $k(\mathbf{v}_i, \bar{\mathbf{v}})$.

$$k(\mathbf{v}_i, \bar{\mathbf{v}}) = w_b e^{-0.5g_i} e^{-2d_i} \quad (9)$$

with

$$g_i = \left\| \frac{\theta(\mathbf{v}_i, \bar{\mathbf{v}})}{W} \right\|^2, \quad d_i = \left\| \frac{s(\mathbf{v}_i)}{s(\mathbf{v}_i^0)} \right\|^2, \quad (10)$$

where $s(\mathbf{v}_i)$ gives the length of rest path from i 's current position to the target and $s(\mathbf{v}_i^0)$ returns the length of i 's path from its initial position \mathbf{v}_i^0 to the target.

- Step 3): The new mean can be computed using the weighted mean of the density in the window

$$\bar{\mathbf{v}}_{new} = \frac{\sum_{i \in \mathcal{S}} \varpi(\mathbf{v}_i, \bar{\mathbf{v}}) k(\mathbf{v}_i, \bar{\mathbf{v}}) \mathbf{v}_i}{\sum_{i \in \mathcal{S}} \varpi(\mathbf{v}_i, \bar{\mathbf{v}}) k(\mathbf{v}_i, \bar{\mathbf{v}})}. \quad (11)$$

- Step 4): Shifting the center of window to $\bar{\mathbf{v}}_{new}$ and go back to Step 2 if the terminal condition has not been met.

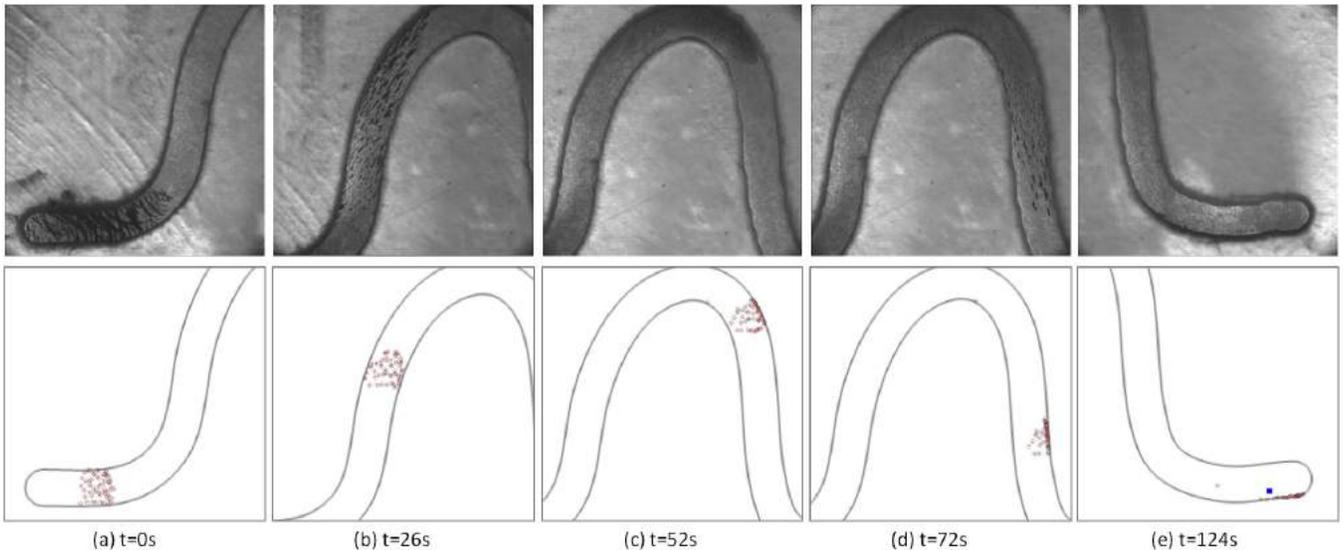


Fig. 7. The comparisons of experiments and our simulation at different time frames.

Here, a hybrid terminal condition is used. Iteration stops when the angle different between the current and the previous \bar{v} is less than 0.01 or the algorithm has run more than 100 iterations. In all our experiments, the mean-shift algorithm converges with 20 iterations. After that, the resultant \bar{v} is employed as the actuating direction to transport all agents at this time frame.

In Eqs.(9) and (10), the component e^{-2d_i} makes the leading agent that is closer to the target have a much greater weight, which helps the swarm moving to the target in the least possible time according to the ‘leader-follower’ law in crowd simulation [21]. In practice, micro-robots can easily stick to the boundary of a tunnel, which can slow down the total time of swarm. To overcome this difficulty, a boundary weight w_b is introduced in Eq.(9). When an agent is located inside a boundary grid, a large value is assigned to w_b (e.g., $w_b = 10$ in all our examples). And $w_b = 1$ is used for all other cases.

V. EXPERIMENTAL TESTS

A. Hardware

Our experimental tests are conducted on an electromagnetic device as shown in Fig.2. In this device, a three-axis Helmholtz electromagnetic coils setup was used to generate a magnetic field for actuating the magnetic micro-robots. Maximal strength of the magnetic field can reach 10mT with the minimal step as 0.1mT. Each pair of coils is connected to an amplifier and they are controlled by a Sensory 826 card. The magnetic control and the micromanipulation experiments were conducted in a 30mm \times 30mm workspace. When changing the electric currents applied to the electromagnetic coils, direction of the forces applied in the working envelope is also adjusted accordingly [1]. With the help of this numerically controlled setup, we are able to program the time-series sequence of actuating field’s directions. Note that we have not integrated visual feedback

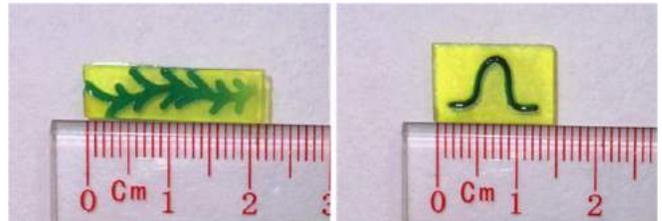


Fig. 8. The channels used in our experimental tests.

into this swarm steering system of micro-robots, which will be considered in our future work.

B. Particle Aggregation

Aggregated nano-particles are employed as micro-robots in our test as the nano-particles themselves are too small to be observed under the microscope with a magnification of 20 \times . It is not an easy job to control the shapes of aggregation. We first apply a 10mT-strength uniform magnetic field, under which magnetic nano-particles are attracted to each other to form micro-chains when they are close enough. Meanwhile, an oscillating field (strength: 10mT) with its normal direction perpendicular to the experimental plane is also applied to prevent the micro-chains being stuck to the bottom of test environment (e.g., the channels shown in Fig.8). In this process, magnetic fields with high strength were applied to oscillate micro-chains in the experimental plane to attract the remaining nano-particles to further aggregate.

After starting to actuate the transportation, the micro-chains will continue to attract each other if they are too close. Therefore, the strength of magnetic field is changed to 3mT to reduce the magnetic interaction force between two tumbled micro-chains. However, the strength cannot be further reduced as the strength of a 2mT magnetic field is too low to transport the micro-chains following the computed actuating directions.

C. Physical Verification

In this work, two channels as shown in Fig.8 are used to verify the steering strategy of micro-robotic swarm. The channels are filled with DI water. And the micro-chains aggregated from nano-particles are placed at the initial locations according to the computational tests by using a needle tube. After applying the sequence of dynamic magnetic fields, the micro-chains responded immediately. Pairs of states in simulation and experiment are shown in Fig.7, where the experiment is similar to simulation. By using our steering strategy, the swarm of micro-chains can pass through the channel with little loss. The final state is shown as Fig.7(e), and the reaching rate is about 76.5%. Another test taken in a branch-shaped channel has been included in the attached video, in which the reaching rate is over 90. A major part of loss is due to the friction between the chains and the substrate, where nano-particles are stuck onto the substrate and cannot make any locomotion. To estimate the reaching rate, we count the number of chains at the target position and those left in other places.

VI. CONCLUSION AND DISCUSSION

In this paper, we control the micro-robotic swarm by using dynamic fields that can only generate a single actuating direction at each time frame. All the robots are controlled to move from their initial positions to the target position in a short time while avoiding obstacles. A two-level planning framework is presented in this paper, which is general to steer micro-robotic swarm by any type of dynamic fields. A vector field is first computed for global navigation, followed by a mean-shift based steering method to compute the optimal actuating direction at each time-step. Our approach offers a simple, general and efficient method to control the motion of a large group of micro-robots in arbitrary shaped environments. Experimental results have been taken at the end of this paper to verify the performance of our approach. Results can also be found in the accompanying video.

Although it is promising, the approach presented in this paper is still preliminary. The work can be improved in quite a few aspects. First, vision feedback can be added into the platform to track the real positions of micro-robots so that the actuating forces can be generated in a more precise way. Second, it will be more interesting if the steering is taken in a dynamic environment, i.e. with dynamic targets or in an environment that is not completely known in advance. In such a scenario, the governing field and the sequence of directions must be computed in an interactive rate. As a result, the dynamic target can be used to serve as a leader to attract the followers (i.e., micro-robots) to follow the motion of a leader.

REFERENCES

- [1] S. Tasoglu, E. Diller, S. Guven, M. Sitti, and U. Demirci, "Untethered micro-robotic coding of three-dimensional material composition," *Nature Communications*, no. 5, 2014.
- [2] L. Zhang, J. J. Abbott, L. Dong, B. E. Kratochvil, D. Bell, and B. J. Nelson, "Artificial bacterial flagella: Fabrication and magnetic control," *Applied Physics Letters*, vol. 94, no. 6, 2009.
- [3] E. B. Steager, M. S. Sakar, C. Magee, M. Kennedy, A. Cowley, and V. Kumar, "Automated biomaniipulation of single cells using magnetic microrobots," *The International Journal of Robotics Research*, vol. 32, no. 3, pp. 346–359, 2013.
- [4] S. Martel, "Targeted delivery of therapeutic agents with controlled bacterial carriers in the human blood vessels," in *2006 Bio Micro and Nanosystems Conference*, 2006.
- [5] S. Floyd, C. Pawashe, and M. Sitti, "An untethered magnetically actuated micro-robot capable of motion on arbitrary surfaces," in *International Conference on Robotics and Automation*, 2008, pp. 419–424.
- [6] A. Solovev, S. Sanchez, and O. G. Schmidt, "Collective behaviour of self-propelled catalytic micromotors," *Nanoscale*, vol. 5, pp. 1284–1293, 2013.
- [7] W. Wang, L. A. Castro, M. Hoyos, and T. E. Mallouk, "Autonomous motion of metallic microrods propelled by ultrasound," *ACS Nano*, vol. 6, pp. 6122–6132, 2012.
- [8] M. Ibele, T. Mallouk, and A. Sen, "Schooling behavior of light-powered autonomous micromotors in water," *Angewandte Chemie International Edition*, vol. 48, no. 18, pp. 3308–3312, 2009.
- [9] M. J. Kim, S. Brigandi, A. A. Julius, and M. J. Kim, "Real-time feedback control using artificial magnetotaxis with rapidly-exploring random tree (rrt) for tetrahymena pyriformis as a microbiorobot," in *International Conference on Robotics and Automation*, 2011, pp. 3183–3188.
- [10] Y. Ou, D. H. Kim, P. Kim, M. J. Kim, and A. A. Julius, "Motion control of magnetized tetrahymena pyriformis cells by a magnetic field with model predictive control," *International Journal of Robotics Research*, vol. 32, no. 1, pp. 129–139, 2013.
- [11] Y. Ou, P. Kang, M. J. Kim, and A. A. Julius, "Algorithms for simultaneous motion control of multiple t. pyriformis cells: Model predictive control and particle swarm optimization," in *International Conference on Robotics and Automation*, 2015, pp. 3507–3512.
- [12] X. Wang, S. Chen, M. Kong, Z. Wang, K. D. Costa, R. A. Li, and D. Sun, "Enhanced cell sorting and manipulation with combined optical tweezer and microfluidic chip technologies," *Lab Chip*, vol. 11, pp. 3656–3662, 2011.
- [13] T. Vicsek and A. Zafeiris, "Collective motion," *Physics Reports*, vol. 517, no. 3, pp. 71–140, 2012.
- [14] J.-C. Latombe, *Robot motion planning*. Springer Science & Business Media, 2012, vol. 124.
- [15] M. Botsch and L. Kobbelt, "An intuitive framework for real-time freeform modeling," *ACM Trans. Graph.*, vol. 23, no. 3, pp. 630–634, 2004.
- [16] M. Wardetzky, S. Mathur, F. Kälberer, and E. Grinspun, "Discrete laplace operators: No free lunch," in *Proceedings of the Fifth Eurographics Symposium on Geometry Processing*, ser. SGP '07, 2007, pp. 33–37.
- [17] D. Delling, P. Sanders, D. Schultes, and D. Wagner, "Engineering route planning algorithms, algorithmics of large and complex networks: Design, analysis, and simulation," 2009.
- [18] K. Fukunaga and L. D. Hostetler, "The estimation of the gradient of a density function, with applications in pattern recognition," *IEEE Transactions on Information Theory*, vol. 21, no. 1, pp. 32–40, 1975.
- [19] Y. Cheng, "Mean shift, mode seeking, and clustering," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 17, no. 8, pp. 790–799, 1995.
- [20] D. Comaniciu and P. Meer, "Mean shift: A robust approach toward feature space analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, 2002.
- [21] C. W. Reynolds, "Steering behaviors for autonomous characters," in *Game developers conference*, vol. 1999, 1999, pp. 763–782.