Path Planning and Tracking for an Underactuated Two-Microrobot System

Mohammad Salehizadeh and Eric Diller

Abstract—We propose a robotics method to independently control the position of multiple magnetic microrobots (agents) through their local magnetic interactions in close proximity. Utilizing a rapidly-exploring random tree (RRT) planner, we generate a trajectory which is then followed by an optimization-based controller (OBC), while avoiding obstacles. The method generates a path that obeys the governing unique nonholonomic actuation constraints in the agents' motions, and has potential to scale to more agents. This is the first work of its kind to introduce a method for autonomous navigation of multiple magnetic microrobots in close proximity with obstacle avoidance. The ability to independently control the motion of multiple magnetic microrobots in this manner makes the method useful for potential applications such as targetted drug delivery and medical diagnostics. We prove that the two-microrobot system studied here is small-time locally controllable (STLC). We conduct two practical demonstrations to verify the ability of the RRT-based navigation applied to a pair of magnetic microrobots moving in a microchannel: 1) multi-agent micro-factory, and 2) team-targeted cargo delivery. An average tracking error of around one-fifth of the agents' body-length is achieved for the control of agents' positions, while avoiding collisions between agents.

Keywords: multi-agent control at small scales, nonholonomic motion planning, RRT, underactuated robotics, collision avoidance, micro-factory, microgrippers.

I. INTRODUCTION

TEAM control of magnetic microrobots offers a great opportunity to perform tasks in parallel that would be difficult to achieve with a single microrobot. Nonetheless, the need to design real-time planning and tracking of a collision-free reference trajectory becomes crucial in navigating a team of magnetic microrobots in cluttered and constrained environments, such as the human body or microfluidic channels [1]. The key attribute to be realized is that the synthesized trajectory for the multi-agent system must obey the physical and actuation constraints associated with the system.

In the context of motion planning, path planning of constrained robotics systems have been extensively studied. Hong et al. [2] recently proposed a 3D path planning method to steer flexible needles along curved paths in deep brain stimulation procedures. They used an RRT approach combined with the reachability-guided strategy by taking into account two types of constraints: 1) constraints coming from anatomical obstacles in neurosurgery, 2) physical constraints dictated by flexible needle kinematics. Liu et al. [3] used an informed RRT* path planner followed by a sliding mode controller to

navigate a helical microswimmer. They used a neural network technique to set up actuation-movement relationship without a state observer. This AI unit computes the raw nominal control inputs in the absence of disturbance through the feedforward loop. The sliding mode controller then adaptively fine-tunes the nominal control inputs by processing the closed-loop feedback data on the state error, based on the force model of insertion or push. Xie et al. [4] described a method to control a snakelike magnetic microrobot swarm by changing the parameters of a rotating magnetic field through a genetic algorithm. They achieved tracking via the Meanshift algorithm. Ongaro et al. [5] applied RRT-GoalZoom policy integrated with a closed-loop magnetic pulling force control and thermally-induced open and closure states to perform pick-and-place of biological materials using a single microgripper. Dong and Sitti [6] utilized a planner that could plan different motion primitives for grasping, and compensate the uncertainties in the motion of a single microgripper. However, these studies were done only to control a single magnetic microrobot or a single swarm-like particle [7].

The prime motivation for this work is that independent navigation of multiple magnetic microrobots in close proximity under a global driving signal with obstacle avoidance has not been shown. When pairs of magnetic agents get close to each other, they tend to stick together irreversibly by magnetic attraction. Most studies on multi-microrobot control ignore this issue and assume that inter-agent magnetic fields are small in comparison with the driving actuation field strength, with the resulting requirement that the agents be kept far apart from each other [8]. This constraint on minimum spacing limits the ability of teams of agents from working closely together, and the motion of each agent is often assumed to be holonomic. However, in close proximity, strong inter-agent magnetic forces appear among the magnetic agents, which makes the agents' equation of motion nonlinear. The system is underactuated given the assumptions made on homogeneous field as the only input, and the motion of agents obey nonholonomic constraints arising from actuation. The constraint can be formulated mathematically, but is not as kinematically intuitive as for other constrained robotics systems. The need to solve this existing problem modelled by unintuitive nonholonomic actuation constraints motivates the use of a sampling-based planner. Therefore, this paper is needed to show the implementation details and challenges with this approach for such multi-agent constrained systems.

The goal of this research is to achieve collision-free autonomous navigation of two magnetic microrobots in close proximity such that the agents' motions obey the governing unique nonholonomic actuation constraints. We benefit from an RRT for path planning, and use the OBC developed in our prior

work [9], [10] for path tracking. An RRT planner is designed for efficiently searching nonconvex high-dimensional spaces that involve state differential constraints similar to our multi-agent system. The study presents the following contributions:

1) For our past controllers, including the PI-controller [11], [12] and the OBC [10], the user had to manually program a reference trajectory ahead of an experiment, which was not necessarily feasible. In this paper, we design an RRT path planner to autonomously generate a reference path from initial to goal configurations, where the RRT guarantees that the derived path has satisfied the system's nonholonomic actuation constraints in real time. Like parallel parking a car, where direct sideways motion is constrained, in our system for some configurations there is no actuation input angle or magnetic gradient term to generate a desired local or external magnetic force. As a solution, we achieve these prohibited motions indirectly with an RRT path search by approximating them with a series of forward-backward and turning maneuvers.

We assume all magnetic microrobots are simultaneously aligned with the applied magnetic field. Fig. 1 shows the physics-based simulation of the local achievable motion forces for a pair of magnetic agents along a planned path by our developed RRT. In opposite to more than two agents for which impossible motion directions exist, for two agents every motion direction seems possible spanning only the perimeter of an ellipse but the magnitude of motion is constrained. As the applied uniform field oscillates, the radial and transverse inter-agent forces respectively may get attractive or repulsive, and clockwise or counter-clockwise. In the RRT planner design, one can penalize a separation that is large but requires a large effort to rotate. This policy is realized by incorporating the separation(s) and pair heading(s) in the distance cost when assigning the nearest reachable node, or in the stop criterion.

2) We prove by using Lie bracket analysis that the two-microrobot system under our control design is STLC. This result is the beginning of future studies to determine the STLC formation configurations for a generic number of magnetic microrobots (agents) using a single control input. Once identified, one can consider those formations as the goal configuration candidates in the RRT-based navigation.

This paper shows how to prescribe the motion of two magnetic agents in close proximity using an RRT, and introduce a potential solution for more agents. We describe the 2D kinematics of agents in Section II. Sections III and IV introduce the motion control of two magnetic microrobots and prove its STLC property, respectively. Next, we investigate the feasibility of the proposed navigation method in Section V. Section V-D describes the application of RRT-based navigation to solve a multi-agent micro-factory problem. Section V-E further incorporates the idea to run a team-targeted cargo delivery demo. This article is concluded in Section VI.

II. BACKGROUND

This section presents the kinematics of magnetic agents operating close to each other under the influence of a homogeneous quasi-static magnetic field. A "homogeneous quasi-static field" describes a field that is spatially uniform and constant over short time periods [10].
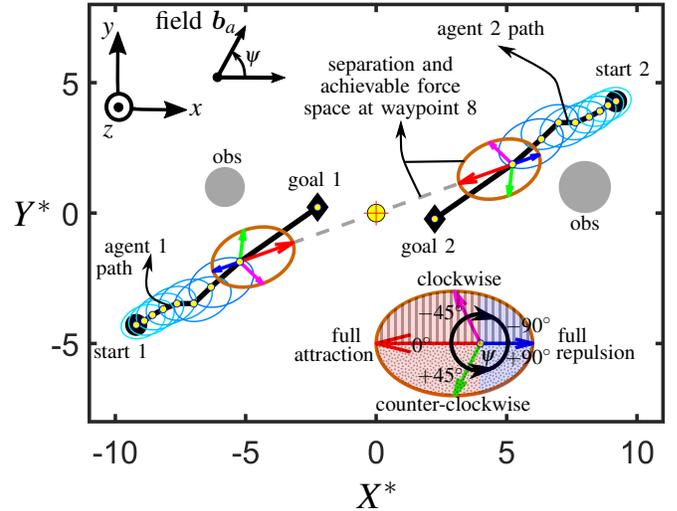
Fig. 1: Visualization of the achievable inter-agent forces centered at the waypoints along a reference trajectory planned by the RRT for a pair of magnetic agents. For this particular simulation, the center-of-mass (COM) of the team of agents is fixed at the origin. The attractive and repulsive radial force domains are shaded in red and blue, respectively. The clockwise and counter-clockwise transverse force domains are shaded in stripes and polka-dots, respectively. Due to inter-agent kinematic constraints, the two microrobots have to follow a feasible path represented in black. The uniform magnetic field denoted by b_a rotates in 2D by an angle ψ. By sweeping the ψ from $-90°$ to $+90°$, possible motion directions only span the perimeter of an ellipse. The achievable force space at the 8th waypoint is highlighted in brown and marked with the principal inter-agent forces. Based on the resulting actuation constraint [11], the maximum transverse inter-agent force (in green and pink arrows) that can be generated is half of the maximum radial inter-agent force (in red arrow). Given the difficulty to display the sequence of achievable forces due to the rapid force variations with respect to separation, the ellipse-like force spaces are enlarged and logarithmically scaled.

A. Magnetic forces between a pair of magnetic agents

The kinematics are presented in our previous work [11], but the key details are repeated here. For the magnetic dipole pair (i, j) moving in a 2D plane, we consider local cylindrical coordinates $(\hat{e}_r, \hat{e}_\phi, \hat{e}_z)$ defined with respect to the pairwise separation vector $r_{ij} = P_j - P_i$, where P_j and m_j represent the position and magnetic moment of the jth agent, respectively. Under a quasi-static magnetic field b_a (see Fig. 2), the radial, transverse, and altitude components of the local magnetic force exerted on the agent j by the agent i can be written as

$$f_{r_{ij}} = \frac{\Omega}{r_{ij}^4}[1 - 3\cos^2(\alpha)\cos^2(\psi)], \tag{1a}$$

$$f_{\phi_{ij}} = \frac{\Omega}{r_{ij}^4}[\cos^2(\alpha)\sin(2\psi)], \text{ and} \tag{1b}$$

$$f_{z_{ij}} = 0. \tag{1c}$$

Here $\Omega := \frac{3\mu_0 m_i m_j}{4\pi}$ is the force constant. μ_0 is the permeability of free space, ψ is the local in-plane (yaw) control input angle made between the projection of the applied field b_a on the motion plane and the separation vector r. Similarly, $\psi_G = \psi + \phi$ is the in-plane control angle in the global coordinate. The out-of-plane (pitch) control input angle made between the field b_a and its projection on the motion plane is denoted by α. As a proof of concept, we constrain the magnetization to sweep in only the horizontal plane, i.e.,

$\alpha = 0°$. We also employ the liquid-air interface as the physical constraint which restrains the elevation force $\boldsymbol{f}_z = 0$. The relationship between the forces and the magnetic moment of the agent acting as the field source (either $\boldsymbol{m}_i$ or $\boldsymbol{m}_j$) is nonlinear [10], [13]. The main assumption behind this work is that all magnetic moments on the mobile microrobot agents are simultaneously aligned with the applied field. Throughout this article, all distance variables are normalized by 2R that denotes the body diameter of agents if modelled as a sphere. These normalized distances are denoted by an asterisk. Based on the Stokes fluid drag model, the nonholonomic equation of constraint for the local states r and ϕ is written as,

$$\dot{r} + r\dot{\phi} = \frac{-\Omega}{\sigma r^4}[1 - 3\cos^2(\psi) + \sin(2\psi)], \qquad (2)$$

where σ is the fluid drag constant.

B. Nonholonomic car-like robot motion model for a two-agent configuration in 2D

One can model the pair of agents as a car-like robot and benefit from existing methods to study the system. The path taken by two agents are expected to be symmetrical though depends on both their initial positions and the spatial distribution of the total magnetic potential energy. For actuation, a uniform magnetic field is steered to adjust the relative position of agents and a weak magnetic field gradient is superimposed to pull around the centroid of the team of agent.

The following equation represents the affine-control model S_{global} for a two-agent configuration. By considering the angular velocity of the input field angle $u = \dot{\psi}$ as the model input, one can write down the non-inertial system kinematics (see Fig. 2).

$$(S_{\text{global}}): \begin{bmatrix} \dot{x}_c \\ \dot{y}_c \\ \dot{r}_{12} \\ \dot{\phi}_{12} \\ \dot{\psi} \end{bmatrix} = \underbrace{f(\boldsymbol{x})}_{\text{drift}} + \underbrace{g(\boldsymbol{x})u}_{\text{control input}} = \begin{bmatrix} f_x(x_c) \\ f_y(y_c) \\ f_{r_{12}}(r_{12}, \psi) \\ f_{\phi_{12}}(r_{12}, \psi)/r_{12} \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \dot{\psi}.$$
$$(3)$$

The drift and control vector fields are denoted by f and g that indicate directions along which the states $\bar{\mathrm{x}}_{\text{global}} = [x_c, y_c, r, \phi]^\top$ can move [15]. The physical intuition of this modelling is that with $\dot{\psi}$, we only have control on the steering velocity of each agent, and the inter-agent forces $\boldsymbol{f}_{r_{12}}$ and $\boldsymbol{f}_{\phi_{12}}$ are the local drift terms to our system. The external magnetic pulling forces along the x and y axes in the global frame are denoted by $\boldsymbol{f}_x$ and $\boldsymbol{f}_y$ that are constrained in our case generated by a tri-axial Helmholtz coil system [10].

C. Motion actuation constraints

In the presented system, underactuation comes from lack of control inputs, and nonholonomic motion constraints stem from the actuation via the inter-agent forces and singular magnetic pulling. Our system introduces three motion actuation constraints as follows:

1) One constraint called "proximity obstacle" describes the limited range of separations that can be actuated stably in close proximity $r_{\min} < r < r_{\max}$. This range conveys the electromagnetic communication radius of our agents.
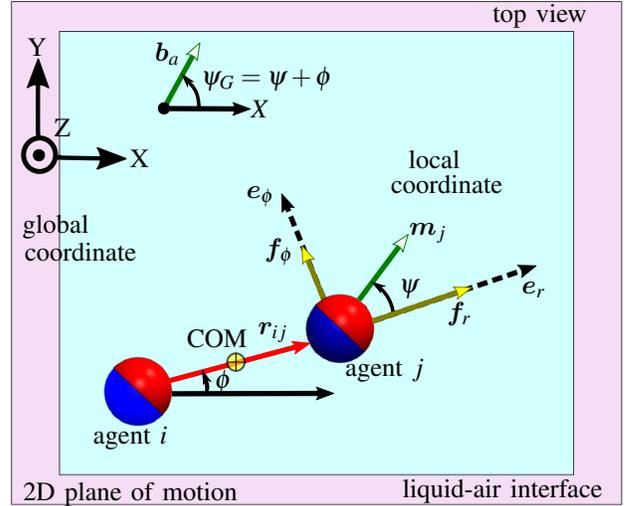
Fig. 2: 2D kinematics of two magnetic agents under the influence of a uniform magnetic field $\boldsymbol{b}_a$. The definitions of relevant parameters are given in the global and local coordinate frames. The separation vector pointing from agent i to j and the pair heading are denoted by $\boldsymbol{r}_{ij}$ and ϕ, respectively. The radial and transverse coordinates are shown by $\hat{e}_r$ and $\hat{e}_\phi$ along which the radial and transverse forces appear parallel and perpendicular to the separation vector.

2) Another constraint belongs to the nonlinear actuation map between the input magnetic field's orientation ψ and the pairwise local states of the system $\bar{\mathrm{x}}_{\text{local}} = [r, \phi]^\top$.
3) Since we are using an underactuated coil system, the magnetic pulling that applies to control the center-of-mass $\bar{\mathrm{x}}_{\text{COM}} = [x_c, y_c]^\top$ will impose singularities on the agents' motions [10].

Any path that does not satisfy these rules is rejected during each RRT path sampling routine.

III. Motion Control of Two Magnetic Microrobots

This section explains our multi-robot planning and tracking of magnetic microrobots as shown in Fig. 3(A). The navigation system is comprised of three subsystems: 1) a path planning unit (in yellow) which generates an obstacle-avoidant reference trajectory by finding suitable open-loop control inputs denoted by ψ_{ol}, 2) a path tracking unit (in pink) to track the derived trajectory and be responsible for the closed-loop feedback tuning of ψ_{ol} through $\Delta\psi_{cl}$, and 3) a visual position feedback unit (in green). We will describe each subsystem as follows.

A. Path planning subsystem

Here we discuss our unidirectional-RRT motion planning implemented for a two-agent configuration in 2D to reach from q_{start} to q_{goal} as structured in Algorithm 1.

Our nonholonomic kinematic motion-planning problem involves finding a continuous path that maps into $\mathcal{C}_{\text{free}} = \mathcal{C} \setminus \mathcal{C}_{\text{obst}}$. At each iteration i, the RRT algorithm samples a state q_{sample} in configuration $\mathcal{C}$-space, finds its nearest neighbor q_{near} in the tree, and computes a feasible control $u = [\psi, f_{x_c}, f_{y_c}]^\top$ that grows the tree toward the sampled state within a particular time step δt_i. If the q_{sample} node falls into an obstacle's convex hull, it will be invalidated and a new sample will be taken. The joint space vector includes team's centroid, separation,
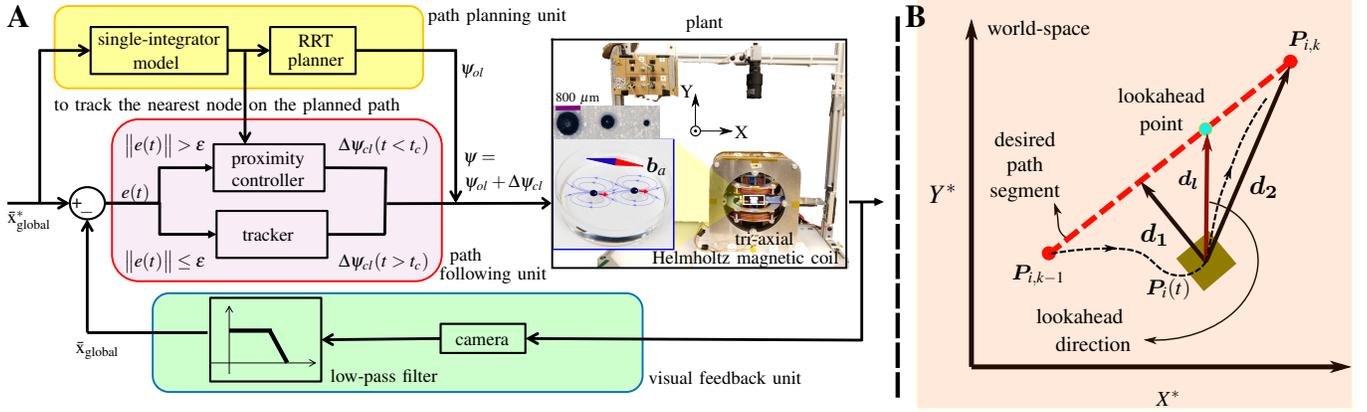
Fig. 3: (A) Multi-robot planning and following block diagram. (B) Pure pursuit (lookahead) algorithm in the X^*Y^*-space shown only for agent i here [14].

Algorithm 1 RRT algorithm

1: Construct a tree as object $\mathcal{T}$ of class RRT to compute a trajectory τ

2: function makeTree($\mathcal{T}, q_{\text{start}}, q_{\text{goal}}$)
3:    $\mathcal{T} \Leftarrow q_{\text{start}}$;
4:    **while** $i <= k$ **do**
5:       $q_{\text{new}} \Leftarrow \mathcal{T}$.BUILD-RRT(); (see algorithm 2)
6:       **if** $\mathcal{T}$.hasReahedGoal() **then**
7:          Break from while loop;
8:       **end if**
9:    **end while**
10:   $\tau \Leftarrow$ computePlan($\mathcal{T}$) via backtracking along $\mathcal{T}$;
11:   return trajectory $\tau = [(q_{\text{start}}, u_{\text{start}}, \delta t_{\text{start}}), \cdots, (q_i, u_i, \delta t_i)]$;
12: end function

Algorithm 2 BUILD-RRT() method

1: Sample a node in the configuration space, find the nearest node on the tree from which an edge will be extended to create a new node—such that kinematic constraints are satisfied and collisions are avoided

2: function BUILD-RRT($\mathcal{T}$)
3:    $q_{\text{sample}} \Leftarrow \mathcal{T}$.randomStateGoalZoom();
4:    $q_{\text{near}} \Leftarrow \mathcal{T}$.nearestNeighbour($q_{\text{sample}}$);
5:    $(q_{\text{new}}, u_{\text{new}}, t_{\text{new}}) \Leftarrow \mathcal{T}$.EXTEND($q_{\text{near}}, q_{\text{sample}}$); (see Algorithm 3)
6:    **if** collisionFree($q_{\text{new}}, u_{\text{new}}$) **then**
7:       $\mathcal{T}$.node $\Leftarrow$ ADD(q_{new});
8:       $\mathcal{T}$.edge $\Leftarrow$ ADD($q_{\text{near}}, q_{\text{new}}, u_{\text{new}}, \delta t_{\text{new}}$);
9:    **end if**
10:   return a valid q_{new}
11: end function

Algorithm 3 EXTEND() method

1: Compute a near optimal new node branching from the nearest node on the tree using Brute-Force optimization such that the physical kinematic constraints are satisfied

2: function EXTEND($\mathcal{T}, q_{\text{near}}, q_{\text{sample}}$)
3:    **while** $i <= k$ **do**
4:       Pick k control inputs randomly and calculate the Euclidean distance error norm for the new node q_{new_i} generated from q_{near}
5:       DISTANCE $\rho(P_{\text{new}_i}, P_{\text{sample}}) = \left\| \tilde{P}_i \right\|^2$, where
6:       $\tilde{P}_i = P_{\text{new}_i} - P_{\text{sample}}$;
7:    **end while**
8:    find the optimal control that minimizes the distance norm
9:    $(u^*, \delta t^*) = \arg \min_{u, \delta t} \; dist = $ DISTANCE $\rho(P_{\text{new}_i}, P_{\text{sample}})$;
10:   $q_{\text{new}}^* = f(q_{\text{near}}, u^*, \delta t^*)$;
11:   return new node $(q_{\text{new}}^*, u^*, \delta t^*)$;
12: end function

---

and pair heading angle denoted by $q = [x_c, y_c, r, \phi]^\top \subset \mathbb{R}^4$. The output of the RRT is a motion trajectory plan $\tau = [(q_{\text{start}}, u_{\text{start}}, \delta t_{\text{start}}), \cdots, (q_{\text{k,goal}}, u_{\text{k,goal}}, \delta t_{\text{k,goal}}))], \; k < T$ where k is the number of steps that construct the trajectory from q_{start} to q_{goal} and the planner is executed every time step δt_i. The RRT continues generating nodes until it finds the desired goal or reaches its limit of time (T) or memory. The control signal which constructs the ith edge of the path is denoted by $u_i = [\psi_i, \; f_{x_i}, \; f_{y_i}]$. In the simulation shown in Fig. 4, the centroid is pulled to the goal, and the input magnetic field angle ψ is modulated to push the relative spacing and pair heading angle toward the goal state.

To extend the tree $\mathcal{T}$ in step 5 of Algorithm 2 (detailed in Algorithm 3), one can use (1) and (3) to compute inter-agent nonlinear force vector fields f_r and f_ϕ for drifts in local states $\bar{x}_{\text{local}} = [r, \; \phi]^\top$ and compute pulling force vector fields f_x and f_y for drift in COM states $\bar{x}_{\text{COM}} = [x_c, \; y_c]^\top$. Both local and global motions will be nonholonomic. The path search occurs in $\mathcal{C}$-space. To choose a random state, we use a standard RRT-GoalZoom policy [16]. Hence, the new random state is
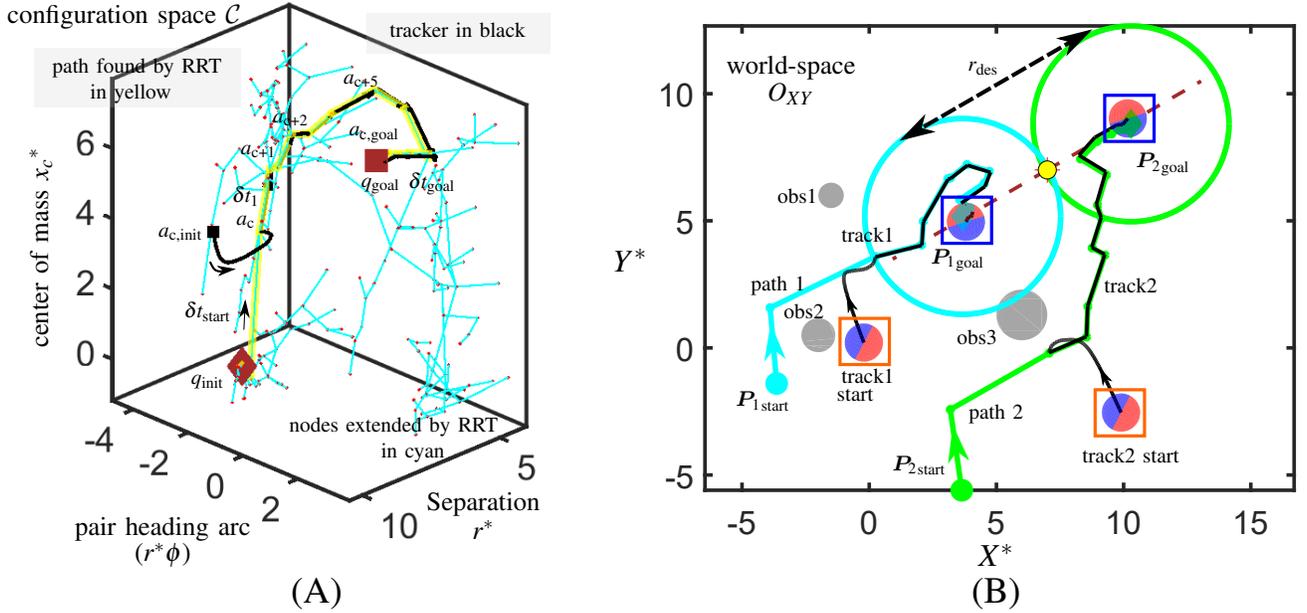
Fig. 4: Simulation result is presented for a collision-free trajectory generation using RRT and its tracking using the OBC, applied to a two-agent configuration in two dimensions (X^*Y^*). (A) The plot shows the corresponding path found by RRT *planner* in the configuration space in yellow. Note that the fourth dimension of the configuration space y_c^* is not shown, as it cannot be pictured in 3D. The brown diamond and square markers represent the start and goal configuration nodes, respectively. Each segment i of the path object is characterized by three members, including a step-time δt_i, an angle ψ_i as the system local control input, and a pulling force f_{ci} as the system global control input. The *tracker* is presented in black with the actual starting configuration represented by $a_{c,\text{init}}$. (B) Associated to the result in (A), the right plot mapped onto the world-space shows the nominal reference trajectory of each agent in cyan and green solid lines, and the actual tracked trajectories in black lines. Initial planned positions are denoted by point circles, goal planned positions with diamonds, and the three obstacles with gray circles. The desired pairwise separation r_{des} is reached when the sketched surrounding circles around agents with radius equal to $0.5r_{\text{des}}$ come into contact with one another. The direction of the desired pair heading angle ϕ is sketched by the brown dashed line. The microrobots are represented as point dipoles in two dimensions. Since the two agents are initially too far, the OBC controller rotate the agents' magnetizations (shown by red-blue poles) to point parallel to their separation vector at the start positions of the tracks (in orange boxes). Once the two agents arrive at goal separation (in blue boxes), their magnetization makes the zero-radial force angle 54.74° with respect to the separation vector.

generated based on a biased coin toss state that chooses a random sample from either a region around the goal or from the whole space, as realized in step 3 of Algorithm 2. Note that since the system must obey nonholonomic constraints, there will be always a region of uncertainty around the goal where drifts do not allow reaching the goal exactly.

B. Path tracking subsystem

The pink block of Fig. 3(A) is responsible for tracking. There are various ways to implement the path tracking subsystem depending on the degree of robustness required for an application. The most general framework that could also be adapted to control more agents is shown in the simulation Fig. 4. At each iteration of the program, Brute-force optimization is utilized as a proximity controller to push the agents toward the nearest nodes on the path τ. To formulate this, let q_{a_c} represent the cth configuration node of the tracker as labelled in Fig. 4(A). As soon as the tracker happens to fall off the path beyond a certain tolerance margin ε i.e., $\|e(t)\| = \|q_{a_c} - q_{i\tau}\| > \varepsilon$, the proximity controller will push the agents back to the nearest node $q_{i\tau}$ on the planned path τ (declared by index i). Once agents arrive sufficiently close to the ith node of τ at time t_c, the tracker as represented in Fig. 3(A), will start driving microrobots in an open-loop manner. In this respect, the tracker steers agents by the stored control input attributes $(u_i, \delta t_i)$, along edges of the path τ from index i increasing to the goal

node's index k. In Section V, a set of experiments are conducted to run the tracker in both open-loop and closed-loop modes.

C. Visual feedback subsystem

The imaging setup and agent fabrication process are similar to those used in our previous work [10] (see Fig. 3 inset). Agents are detected using a camera (FO134TC, FOculus) mounted above, and a computer with custom C++ code finds the center positions of the agents using a Hough-circle transform in the OpenCV library at 60 frames/second.

IV. LIE CONTROLLABILITY ANALYSIS OF A TWO-AGENT CONFIGURATION IN CLOSE PROXIMITY

This section explains how to utilize Lie algebra to check the stability at a given goal configuration. The RRT just returns a trajectory between initial and goal configurations if present. Therefore, we use Lie bracket first to determine the stability at a given configuration. Once the stability is approved, we will use the RRT to accomplish the navigation task. For the purpose of controllability analysis at a given formation, we reduce the order of S_{global} to only deal with the augmented local state space $\bar{\text{x}}_{\text{local}} = [r_{12} \ \phi_{12} \ \psi]$ in this section. One can write down the non-inertial system kinematics as

$$(S_{\text{local}}) : \begin{bmatrix} \dot{r}_{12} \\ \dot{\phi}_{12} \\ \dot{\psi} \end{bmatrix} = \underbrace{f(\text{x})}_{\text{drift}} + \underbrace{g(\text{x})u}_{\text{control input}} = \begin{bmatrix} \frac{\Omega}{r_{12}{}^4}[1 - 3\cos(\psi)^2] \\ \frac{\Omega}{r_{12}{}^5}\sin(2\psi) \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}\dot{\psi} \quad (4)$$
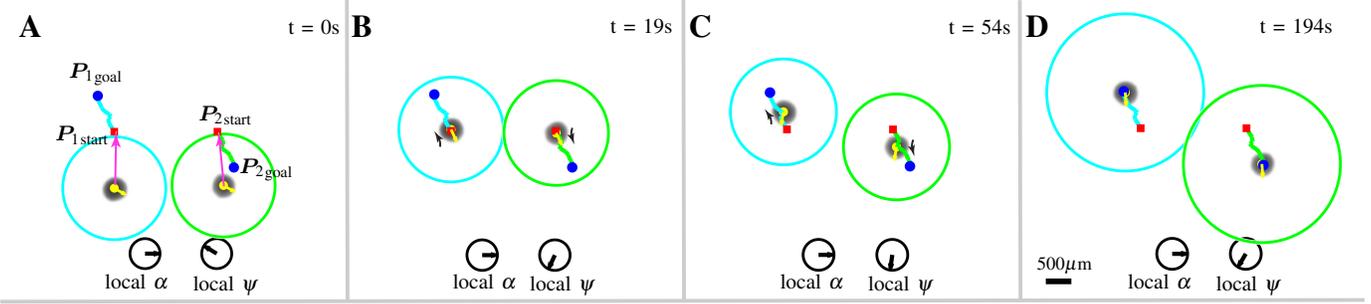
Fig. 5: Sequential experimental snapshots showing open-loop actuation of two magnetic microrobots using the sequence of control input angle ψ generated by the RRT planner from A with start configuration marked by red squares to D with goal configuration marked by blue circles. (A) The proximity controller in the form of OBC steers agents toward the start first. (B) The RRT open-loop tracker (see Fig. 3A) starts at the red squares. (C) The agents are following the track. (D) The agents arrive at the goals in blue circles under the open-loop tracker. Indicator arrows represent the control input angle in the local coordinate.

Note that because an underactuated tri-axial Helmholtz coil system is used, the choice of ψ affects the pulling force $\boldsymbol{f}_x$ and $\boldsymbol{f}_y$ but not vice versa. Therefore, the dynamics of local states will be decoupled from the dynamics of center-of-mass. This point compels us to investigate the stability analysis only for a given geometrical "formation" of the team of magnetic microrobots governed by control input angle ψ and local nonlinear force drifts $f_{r_{12}}$ and $f_{\phi_{12}}$. Let ζ_0 denote the smallest algebra that contains the control vector fields such that $[f, \zeta_0] \subset \zeta_0$, with $[.,.]$ denoting the Lie bracket operator. The associated control distribution $\Delta_{\zeta_0}(\boldsymbol{x}) = \text{span}\{X(\boldsymbol{x}) : X \in \zeta_0\}$ at state point $\boldsymbol{x}$, where ζ_0 is called the strong accessibility algebra. The following lemma gives the necessary condition for nonlinear controllability [15].

Lemma 1. *Let* $\Delta_{\zeta_0} = [g(x), ad_f g(x), \cdots, ad_f^{n-1} g(x)]$. *For a system of order n to be small-time locally controllable (STLC), it should meet the Lie Algebra Rank Condition (LARC), i.e.,* $dim\Delta_{\zeta_0} = n$ *[17].*

Here we check the LARC condition above after applying the Lie bracket twice,

$$
\begin{aligned}
\Delta_{\zeta_0} &= \left[\, g,\ [f,g],\ [f,[f,g]]\, \right] \\
&= \begin{bmatrix} 0 & -\frac{3\sin(2\psi)}{r_{\text{des}}^4} & 0 \\ 0 & -\frac{2\cos(2\psi)}{r_{\text{des}}^5} & -\frac{10\ (1+\cos^2(\psi))}{r_{\text{des}}^{10}} \\ 1 & 0 & 0 \end{bmatrix}
\end{aligned} \tag{5}
$$

The control input and drift vector fields are denoted by $g = [0,\ 0,\ 1]^\top$ and $f = [f_r,\ f_\phi/r,\ 0]^\top$, respectively, associated with the local states $\bar{\text{x}}_{\text{local}} = [r,\ \phi,\ \psi]^\top$. It is evident from (5) that the Lie bracket tree matrix Δ_{ζ_0} applied to the local states has full rank of 3 for any local input angle $\psi \neq 0, 90°$, thus the two-agent configuration is STLC; in other words, the control distribution associated with a single-pair system spans the whole space $\mathbb{R}^3$; this can be interpreted by the fact that at the boundary angles $0°$ and $90°$, the radial force is maximum but no transverse force can be generated at all, hence no control on the ϕ state. It is not possible to arbitrarily make both $\boldsymbol{f}_r$ and $\boldsymbol{f}_\phi$ forces zero with a single control input angle ψ. However, at a desired configuration called dynamic equilibrium, our controller can dynamically make $\boldsymbol{f}_r$ zero by picking $\psi = \pm54.74°$ and flip the sign of the ψ angle to reverse the sign of $\boldsymbol{f}_\phi$ while crossing zero. This observation offers a strong analogy with

the STLC property of the system. The reason is that a control solution already exists that is capable to make all drift forces zero for a small time and hold two magnetic agents in place. Now that the STLC of the two-agent configuration is approved, let's use RRT to accomplish the navigation task.

V. EXPERIMENTAL RESULTS

This section presents a set of experimental results that aim to show the efficacy of the proposed RRT-based navigation to perform the stabilization of two magnetic agents to their desired positions in a 2D plane. We used glycerol-water solution with approximate viscosity of 30mPa.s for the liquid.

A. Open-loop actuation

In the first experiment illustrated in Fig. 5, we validate the predictability of our kinematic model. At the beginning, the proximity controller realized by the OBC is employed to steer the agents to the nearest node (see A). Then, the tracker unit (see Fig. 3A) is involved to smoothly steer agents from start to goal, in an open-loop manner, by following the control input sequence returned by the RRT. Video is available in the supplementary materials. The result of this experiment indicates that the kinematic model (3) is valid and it can be used as an *a priori* model for feedback control systems. A potential feature of this navigation scheme is to offer a generic solution for more than two agents. In this respect, based on the distance error from the path at each iteration, the algorithm decides to interchange between the proximity controller and the tracker. The proximity controller can be best implemented by the OBC or an informed-RRT* to be optimal. The tracker, on the other hand, can be formed by the RRT path generator to be run in an open-loop fashion until multiple agents arrive at the goal.

B. Closed-loop actuation

We performed the experiments in Sections V-D and V-E through closed-loop actuation by imposing virtual static obstacles and a goal area. In doing so, both of the proximity controller and the tracker are run by the OBC. The OBC tracker [10] quadratically optimizes the error on each principal local state (separation length or any separation angle) to reach the goal. The optimal radius of the tolerance region ε was found as 5μm for a smooth tracking performance.
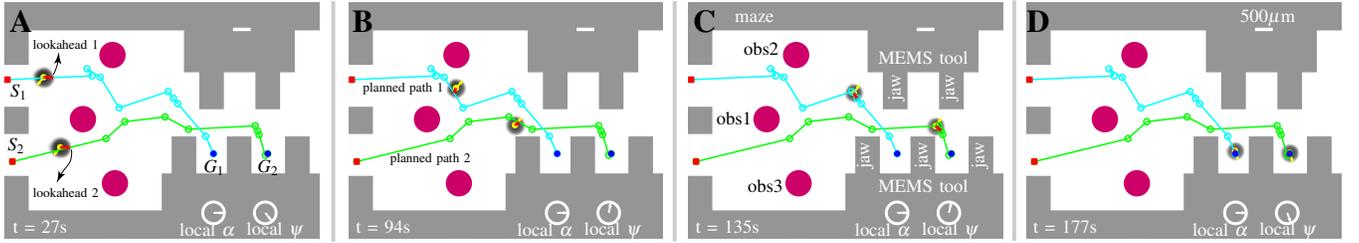
Fig. 6: A representative example of a multi-agent micro-factory. Sequential experimental snapshots shows the closed-loop motion control of two real magnetic microrobots entering a digital microchannel maze. The control task is to navigate the agents from starting points S_1 and S_2 to the test goal points denoted by G_1 and G_2, and hold them in place while avoiding obstacles. The microchannel maze walls and three red circles are treated as obstacle regions. Each agent follows the returned path found by the RRT planner from (A) with start configuration marked by red squares to (D) with goal configuration marked by blue circles. Lookahead directions are shown in red lines connecting current positions of the agents to the moving goals along the planned path (see A). Planned paths are shown in cyan and green for agents 1 and 2, respectively (see B). The two agents are supposed to be navigated and spaced within the MEMS tool chambers and stay in place (see C). The two agents are held in the goal regions and get ready to be tested by the MEMS tool (see D). As example, these agents could represent magnetized cells and the MEMS tool could be either cell lysis beams to lyse the magnetized cell agents, or a gripping squeezer tool connected to sets of electrostatic comb drives to characterize the type of cell agents by measuring their stiffness. See video in the supplementary materials.

C. Pure pursuit waypoint navigation (lookahead)

Multiple magnetic microrobots in close proximity obey nonholonomic motion when tracking a reference path. As such, their motion must be dynamically directed to the reference trajectory. We realized this through the pure pursuit (lookahead) algorithm as described in Fig. 3(B). In the following demos, the lookahead directions are sketched in small red lines [14]. The expected lookahead direction of each microrobot, d_l, given by (6) is a weighted sum of the direction d_2 to the next waypoint P_k and the normal direction back to the path segment d_1.

$$d_l = Ld_1 + (1 - L)d_2. \tag{6}$$

Gain L is to tune how tightly the robot is needed to maintain the path. Without a corrective algorithm such as lookahead, the agents' motions may result in significant rapid jumps between waypoints. Corresponding to each agent by index i, the current position of the agent and previous waypoint position in the world-space are denoted by $P_i(t)$ and $P_{i,k-1}$, respectively. A higher gain L will make the motion faster toward the next goal point but could become unstable. To increment the actual nonholonomic path of the two agents, the algorithm updates the lookahead point moving toward the segment goal. In the planning unit, the step size needs to be tuned based on a logistic function, especially when agents arrive at the goal zone while drifting. Additionally, to ensure agents' motions meet the pairwise kinematic constraint, agents need to traverse the path segments to the end sequentially and in parallel.

D. Demonstration 1: multi-agent micro-factory

The magnetic microrobots can be functionalized on the surface for active drug delivery. Recently, there has been a great deal of investigation around the mechanobiology study of a single cell by measuring its mechanical stiffness inside a chip [18], [19]. The formidable challenge is that the existing approaches have to vigorously deal until a single cell randomly traps into their MEMS tool after manually injecting the cell solution into the chip. To parallelize this process, we employ our proposed RRT-based navigation. The control task is to magnetically automate the process such that multiple representative cell agents are simultaneously steered to the

point of interest, and are held in place for micromanipulation or cell type characterization (see Fig. 6). In this experiment, our RRT planner calculates a collision-free path as soon as the two agents enter the microchannel. Once the path is generated, the OBC tracker is incorporated to converge the agents to the planned paths through the lookahead directions via closed-loop feedback. An RMS tracking position error of around one-fifth of the agents' body-length was achieved.

E. Demonstration 2: manipulation task—targeted cargo delivery using two magnetic microgrippers

This section describes our experimental demo that utilizes the RRT-based navigation for pick-and-place task in 3D using a pair of magnetic microgrippers. In this demo, the microchannel's maze walls and static obstacles were virtually overlaid on the real experiment's imaging plane (see video).

Initially, while the magnetic field is off, the two microgrippers are open (see Fig. 7A). As soon as the field turns on, the grippers get closed and the inter-agent force control (OBC) applies to launch the grippers at the start (see B). Then, the RRT module is called; the agents' paths are synthesized immediately for the top view only (see C). The grippers move pairwisely toward cargoes by tightly tracking the lookahead directions (red lines). The inter-agent alignment occurs to enable the grippers to fly over cargoes (see D). The map switches over to a new set of obstacles. The grippers open, descend, and grasp the cargoes (see E). Afterwards, the grippers face up to do re-grasping of cargoes (see F). The inter-agent control applies back to the loop to bring the agents close enough (see G). By making the RRT planner less biased we chose the ψ to vary rapidly. This allows us to generate a more isotropic 3D pulling force [10]. The subfigure (G) demonstrates that our developed RRT module offers a real-time "replan ability", as the algorithm quickly finds a new set of paths to the final goals upon a request (either as per user's input or in case cargoes drift). The average completion time of our RRT algorithm per 100 trials for the case of the maze above was less than 10 μs. The entire system is controlled through C++ by a single computer with an Intel Core-i7 2.8 GHz processor running Ubuntu Linux. The grippers are holding
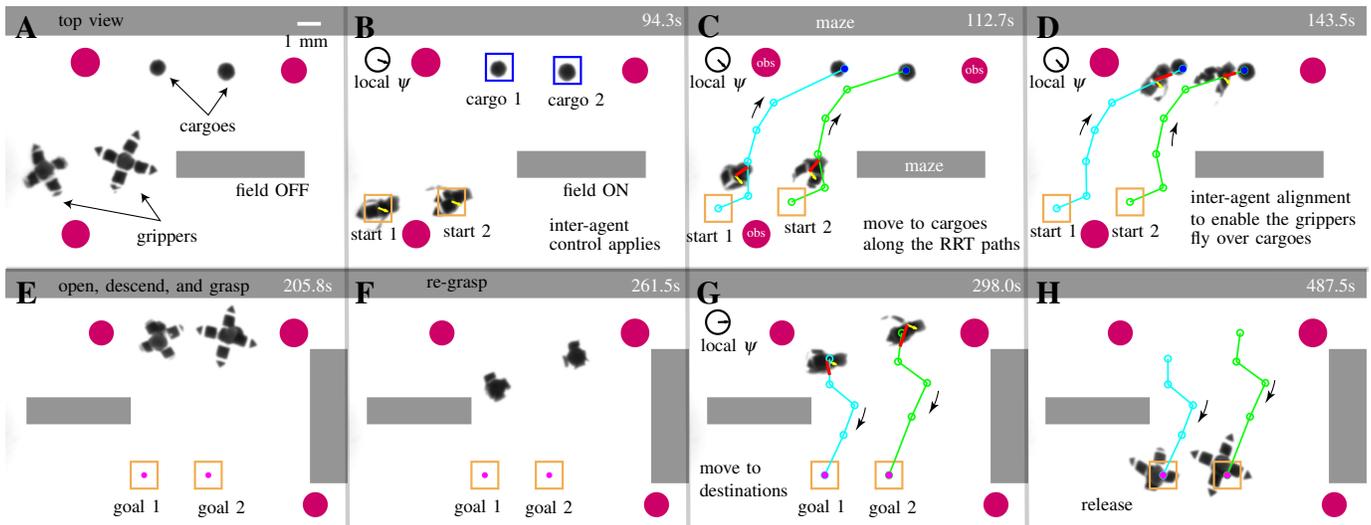
Fig. 7: Sequential experimental snapshots show the collision-free motion control of two magnetic microgrippers entering a digital microchannel maze to pick and place two cargoes in 3D (Only top-view snapshots are shown here; readers are referred to the supplementary video to see a full appearance on both top and side-view cameras). The two microgrippers and two cargoes are submerged in glycerol solution. A sequence of top-view frames of the workspace is shown chronologically from (A) to (H) with the actuation mechanism noted. The local and 3D global control input indicators are provided (see video) with respect to the global pos of the separation vector.

the cargoes and pairwisely moving to the destinations. At the goals, the two microgrippers release the two cargoes (see H).

VI. Conclusions

This work enabled the motion of two magnetic agents in close proximity by coupling an RRT planning algorithm with the OBC while satisfying the unique nonholonomic motion constraints. As such, the system runs autonomously, leading to significant success rates and consistency. Future work will build on the basic idea of this paper by employing an informed-RRT* to accomplish a collision-free navigation for both planning and tracking of three or more agents. The idea could be developed to identify the feasible initial and end configurations for a number of agents. For example, whether a team of agents can reconfigure themselves from an isosceles triangle to a line or vice versa. This analysis could be performed by monitoring the quantitative success rate or the cost of the best paths as a function of iterations averaged over a number of trials. The study could also be useful in the development of medical devices with multiple magnets, by ensuring that the transition paths between different configurations are as safe as possible.

References

[1] Z. Wu, Y. Chen, D. Mukasa, O. S. Pak, and W. Gao, "Medical micro/nanorobots in complex media," *Chemical Society Reviews*, 2020.

[2] A. Hong, Q. Boehler, R. Moser, A. Zemmar, L. Stieglitz, and B. J. Nelson, "3D path planning for flexible needle steering in neurosurgery," *The International Journal of Medical Robotics and Computer Assisted Surgery*, vol. 15, no. 4, p. e1998, 2019.

[3] J. Liu, T. Xu, S. X. Yang, and X. Wu, "Navigation and visual feedback control for magnetically driven helical miniature swimmers," *IEEE Transactions on Industrial Informatics*, vol. 16, no. 1, pp. 477–487, 2019.

[4] H. Xie, X. Fan, M. Sun, Z. Lin, Q. He, and L. Sun, "Programmable generation and motion control of a snakelike magnetic microrobot swarm," *IEEE/ASME Transactions on Mechatronics*, vol. 24, no. 3, pp. 902–912, 2019.

[5] F. Ongaro, S. Scheggi, C. Yoon, F. Van den Brink, S. H. Oh, D. H. Gracias, and S. Misra, "Autonomous planning and control of soft untethered grippers in unstructured environments," *Journal of micro-bio robotics*, vol. 12, no. 1-4, pp. 45–52, 2017.

[6] X. Dong and M. Sitti, "Planning spin-walking locomotion for automatic grasping of microobjects by an untethered magnetic microgripper," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 6612–6618.

[7] D. Jin, J. Yu, K. Yuan, and L. Zhang, "Mimicking the structure and function of ant bridges in a reconfigurable microswarm for electronic applications," *ACS nano*, vol. 13, no. 5, pp. 5999–6007, 2019.

[8] P. Joshi, J. Leclerc, D. Bao, and A. T. Becker, "Motion-planning using rrts for a swarm of robots controlled by global inputs," in *2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*. IEEE, 2019, pp. 1163–1168.

[9] M. Salehizadeh and E. Diller, "Optimization-based formation control of underactuated magnetic microrobots via inter-agent forces," in *Int. Conf. Manipulation, Automation and Robotics at Small Scales*, 2017, pp. 1–5.

[10] M. Salehizadeh and E. Diller, "Three-dimensional independent control of multiple magnetic microrobots via inter-agent forces," *The International Journal of Robotics Research*, p. 0278364920933655, 2020.

[11] M. Salehizadeh and E. Diller, "Two-agent formation control of magnetic microrobots in two dimensions," *Journal of Micro-Bio Robotics*, vol. 12, no. 1-4, pp. 9–19, 2017.

[12] J. Zhang, M. Salehizadeh, and E. Diller, "Parallel pick and place using two independent untethered mobile magnetic microgrippers," in *IEEE Int. Conf. Robot. Autom*, 2018, pp. 1–6.

[13] J. Abbott, E. Diller, and A. Petruska, "Magnetic methods in robotics," *Annu. Rev. Control Robot. Auton. Syst.*, vol. 3, 2020.

[14] R. C. Coulter, "Implementation of the pure pursuit path tracking algorithm," Carnegie-Mellon UNIV Pittsburgh PA Robotics INST, Tech. Rep., 1992.

[15] H. Nijmeijer and A. Van der Schaft, *Nonlinear dynamical control systems*. Springer, 1990, vol. 175.

[16] S. M. LaValle, J. J. Kuffner, B. Donald, *et al.*, "Rapidly-exploring random trees: Progress and prospects," *Algorithmic and computational robotics: new directions*, no. 5, pp. 293–308, 2001.

[17] P. Liljebäck, K. Y. Pettersen, O. Stavdahl, and J. T. Gravdahl, *Snake robots: modelling, mechatronics, and control*. Springer London, 2012.

[18] M. Elhebeary and M. T. A. Saif, "Lessons learned from nanoscale specimens tested by mems-based apparatus," *Journal of Physics D: Applied Physics*, vol. 50, no. 24, p. 243001, 2017.

[19] S. Warnat, H. King, C. Forbrigger, and T. Hubbard, "Polymumps mems device to measure mechanical stiffness of single cells in aqueous media," *Journal of Micromechanics and Microengineering*, vol. 25, no. 2, p. 025011, 2015.