A Multi-Agent Path Planning Strategy with Reconfigurable Topology in Unknown Environments

Hao Sun, Junyan Hu, Li Dai and Boli Chen

***Abstract*— Safety-guaranteed trajectories are important for multi-agent systems to work in an unknown constrained environment. To address this issue, this paper proposes a cooperative path planning strategy for a swarm of agents such that they can achieve a target formation and handle unknown obstacles during complex tasks. By considering the sensing range and agent dimension, a group of artificial potential field functions are designed aiming at enabling agents reconfiguration (e.g., split and merge) for reinforced flexibility. A distributed path planning scheme is then developed to achieve formation tracking while avoiding any potential collisions. Theoretical analysis using the Lyapunov theory is given to guarantee the performance of the system. Finally, numerical simulations are carried out to verify the effectiveness of the proposed algorithm and its superiority against conventional methods.**

I. Introduction

A multi-agent system (MAS) refers to a system consisting of a group of interacting agents that can accomplish complex tasks through intelligent control. Since this concept was first introduced in the early 1980s [1], it has received considerable research attention due to its wide real-world applications. With the recent advancement in robotics, communication, and artificial intelligence technologies, planning and control of MASs has become a popular topic in this field during the past decade.

Motion/path planning (also known as navigation problem) represents one of the most typical MAS control problems, and it appears in a variety of modern engineering applications including connected and autonomous vehicles [2]–[4], mobile and swarm robots [5]–[7] and unmanned aerial vehicles (UAV) [8], [9]. In addition to the traditional methods such as probabilistic roadmap [10], A* [11], Dijkstra [12] and Tangent Bug [13], which are mainly used for single agent path planning. Numerous algorithms have been recently proposed for the MAS including the artificial potential field (APF) [14]–[18], distributed model predictive control (DMPC) [2], and reinforcement learning [19]. It is noteworthy that due to the limited onboard computation power, communication and sensing range, and potential transmission delay [20], most of the available research focuses on distributed or decentralized coordination schemes, whereas centralized methods are used only in case there exists an agent or a central coordinator features sufficient communication and computation resources [21]–[24].

Distributed MAS path planning methods are typically developed based on the leader-follower approach or swarm intelligence. For example, in [2], a DMPC scheme is proposed for a heterogeneous vehicle platoon with a leader vehicle. Provided that there is a spanning tree rooted in the leader, the control feasibility and system stability can be guaranteed. In [25], a multi-robot coordination strategy is proposed based on bearing measurements, where a network of unmanned aerial vehicles can be controlled to achieve a desired formation. A hierarchical plan-and-track framework is created in [5], where the upper layer motion planning is achieved by an APF method, which introduces a potential field to attract the agents towards the targets and to achieve obstacle avoidance. The lower layer involves a DMPC scheme for tracking the trajectories generated by the APF. A robust APF-based trajectory generation algorithm is designed in [16] for a swarm of agents to cope with uncertainties involved in the target information. The APF has also been considered in the 3D space. As shown in [15], a three-dimensional rotating potential field is proposed for path planning of the multi-UAV system. The common strategy to allow multi-agent systems to interact with an unknown environment is to generate the leader's trajectory based on APF and confine followers within the swarm throughout the mission [26], [27]. Although the network connectivity can be maintained, the flexibility may be sacrificed, thus yielding a more conservative path in the presence of densely distributed obstacles, e.g., by pass all obstacles.

Motivated by the aforementioned research progress and limitations, in this paper, we investigate a two-dimensional space path planning problem for a swarm of N agents in a constrained environment. A novel APF-based distributed planning algorithm is designed with capability of flexible reconfigurable formation. As such, a swarm of agents may be split into several sub-groups to accomplish complex tasks. Specifically, topology reconfiguration is reactivated in the event of newly detected obstacles, and thus the overall scheme offers great flexibility in tackling unknown obstacles. In addition, the proposed APF algorithm ensures the boundedness of the resulting velocity trajectory, which can be beneficial for the control layer in practice. The convergence of the algorithm is analyzed under the framework of the Lyapunov-like theory. The benefit of the reconfigurable scheme is shown by comparing it with a single swarm APF-based path planning strategy.

The rest of this paper is organized as follows. In Section

This work has been supported by The Royal Society Grant [IES\R2\212041] and EU H2020-FET-OPEN RoboRoyale [964492].

H. Sun and B. Chen are with the Department of Electronic and Electrical Engineering, University College London, UK, WC1E 6BT (h.sun.20@ucl.ac.uk; boli.chen@ucl.ac.uk).

J. Hu is with the Department of Computer Science, Durham University, UK, DH1 3LE (junyan.hu@durham.ac.uk).

L. Dai is with the School of Automation, Beijing Institute of Technology, Beijing 100081, China (daili1887@gmail.com).

II, the problem formulation is introduced, encompassing the kinematic model with a linearization-based control law and the modeling of obstacles. The Section III proposes the APF-based path planning methodology with capability of flexible reconfigurable topology. In Section IV, the convergence properties of the proposed algorithm are characterized. Simulation results and numerical comparisons are presented in V. Finally, concluding remarks are given in Section VI.

Notation: Let $\mathbb{R}$ and $\mathbb{R}_{>0}$ denote the real set and the positive real set, respectively. Given a vector $x \in \mathbb{R}^n$ and a positive semi-definite matrix $R \in \mathbb{R}^{n\times n}$, $||x||_R = (x^\top Rx)^{1/2}$ denotes the weighted Euclidean norm of x. The difference between two given sets $A \subseteq \mathbb{R}^n$ and $B \subseteq \mathbb{R}^n$ is denoted as $A\backslash B = \{x : x \in A, x \notin B\}$. Given a vector $\eta \in \mathbb{R}^n$ and a positive scalar $\rho \in \mathbb{R}_{>0}$, the closed ball centered at η and of radius ρ is denoted as $\mathcal{B}(\eta, \rho) = \{\xi \in \mathbb{R}^n : ||\xi - \eta|| \leq \rho\}$.

II. Problem Statement

Consider a group of N agents in a two-dimensional space which are initially confined in a swarm with the radius of R_c, and are requested to travel from their initial positions to the predefined target positions without crashing into each other or the obstacles. The motion of each agent is governed by the dynamic model

$$\begin{aligned} \dot{s}_{i,x}(t) &= v_i(t)\cos\theta_i(t) \\ \dot{s}_{i,y}(t) &= v_i(t)\sin\theta_i(t) \\ \dot{\theta}_i(t) &= \omega_i(t),\ \forall i \in \mathbb{A} \end{aligned} \tag{1}$$

where $s_i(t) = [s_{i,x}(t)\ s_{i,y}(t)]^\top$ denotes the position of i-th agent, $\mathbb{A} \triangleq \{1, 2, \ldots, N\}$. $\theta_i(t)$ represents the orientation, and $v_i(t)$ and $\omega_i(t)$ are the velocity and the angular velocity, respectively. Considering the nonholonomic kinematic constraint, the head position of the ith agent can be defined as $p_i(t) = [p_{i,x}(t)\ p_{i,y}(t)]^\top$. In details,

$$\begin{aligned} p_{i,x}(t) &= s_{i,x}(t) + l_i\cos\theta_i(t), \\ p_{i,y}(t) &= s_{i,y}(t) + l_i\sin\theta_i(t) \end{aligned} \tag{2}$$

where l_i represents the distance between the head position and the inertial position. As introduced in [28], the system (1) can be controlled by the feedback linearized kinematic control law:

$$\dot{p}_i(t) = \begin{bmatrix} \dot{p}_{i,x}(t) \\ \dot{p}_{i,y}(t) \end{bmatrix} = \begin{bmatrix} \cos\theta_i(t) & -l_i\sin\theta_i(t) \\ \sin\theta_i(t) & l_i\cos\theta_i(t) \end{bmatrix} \begin{bmatrix} v_i(t) \\ \omega_i(t) \end{bmatrix} = u_i(t) \tag{3}$$

There are M static obstacles in the environment, which are unknown until they are visible to the agents. As shown in Fig. 1, let $R_v > 0$ be the radius of the visual range (based on all onboard radars or cameras of agent i). The sensing range of agent i can be expressed as $\mathcal{B}(p_i(t), R_v)$. Furthermore, the following assumption is introduced to address the collision avoidance against the obstacles in the unknown environment [26], [27].

Assumption 2.1: All obstacles are convex polygons with a sequence of detected obstacle contour points $\{p_l^o(1), p_l^o(2), ..., p_l^o(n)\}$, which are denoted by the red crosses in Fig. 1.

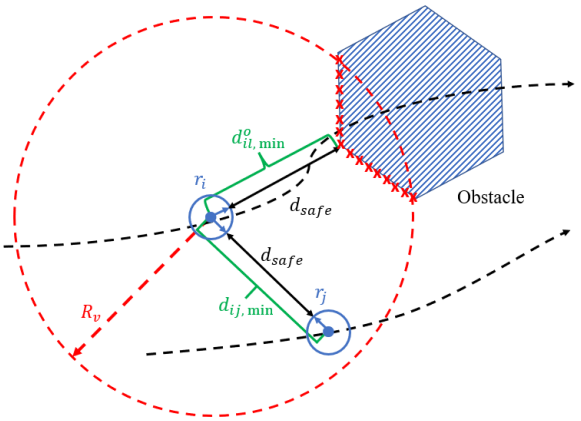


Fig. 1. A demonstration of the collision avoidance constraints.

The objective of the proposed approach is to regulate all the agents so that the MAS can safely converge to a pre-set structure in the steady state by reconfigurable agent formations, such that

$$\lim_{t\to\infty} ||p_i(t) - (p^* + \Delta_i)|| = 0, \quad \forall i \in \mathbb{A} \tag{4}$$

$$||p_i(t) - p_j(t)|| \geq d_{ij,\min}, \quad \forall j \in \mathbb{N}_i(t), \quad \forall t \geq 0 \tag{5}$$

$$||p_i(t) - p_l^o(n)|| \geq d^o_{il,\min}, \quad \forall l \in \mathbb{O}_i(t), \quad \forall t \geq 0 \tag{6}$$

where p^* denotes the center of the target swarm $\mathcal{B}(p^*, R_c)$ whose radius is also R_c. $\mathbb{D} = \{\Delta_i \mid i = 1, 2, ..., N\}$ is a set related to the target formation, where Δ_i represents a vector indicating a fixed offset from p^*. During the initialization stage, p^* and Δ_i are given to corresponding agent i, respectively. $d_{ij,\min} = r_i + r_j + d_{safe}$ is the minimal distance between neighboring agents (see Fig. 1), where r_i, $r_j \in \mathbb{R}_{>0}$ are the radii of agents i and j, respectively, and $d_{safe} \in \mathbb{R}_{>0}$ is the collision avoidance distance between agents. Similarly, the minimum distance between an agent and an obstacle, $d^o_{il,\min} \in \mathbb{R}_{>0}$, follows $d^o_{il,\min} = r_i + d_{safe}$. In addition, $\mathbb{N}_i(t)$ denotes the time-varying neighboring set of agent i with $\mathbb{N}_i(0) = \mathbb{A}\backslash\{i\}$, and $\mathbb{O}_i(t) \subseteq \mathbb{O}$ collects the indices of all obstacles detected by agent i at time t (the set $\mathbb{O}$ includes all obstacles).

III. Path Planning Approach with Reconfigurable Topology

In this section, an APF-based path planning algorithm is designed to find the trajectories of all N agents subject to collision avoidance. In line with the nominal APF algorithm, we propose the distributed control law for (3)

$$u_i(t) = \nabla U_{i,att}\big(p_i(t)\big) + \nabla U^a_{i,rep}\big(p_i(t)\big) + \nabla U^o_{i,rep}\big(p_i(t)\big) \tag{7}$$

where $U_{i,att}\big(p_i(t)\big) : \mathbb{R}^2 \to \mathbb{R}$ is the attractive potential field, $U^a_{i,rep}\big(p_i(t)\big) : \mathbb{R}^2 \to \mathbb{R}$ and $U^o_{i,rep}\big(p_i(t)\big) : \mathbb{R}^2 \to \mathbb{R}$ are the repulsive potential field against neighbouring agents and obstacles, respectively. $\nabla U_{i,att}\big(p_i(t)\big)$, $\nabla U^a_{i,rep}\big(p_i(t)\big)$

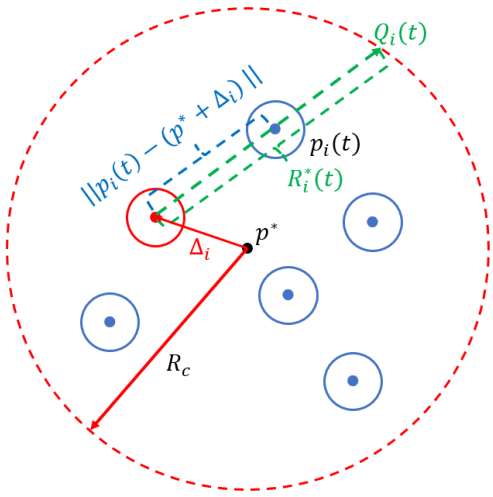


Fig. 2. Schematic diagram about normalized radius $R_i^*(t)$ design.

and $\nabla U^o_{i,rep}\big(p_i(t)\big)$ are the gradients with respect to $p_i(t)$. More specifically, the attractive potential field follows

$$U_{i,att}(p_i(t)) = \begin{cases} \frac{1}{2}k_{att}||p_i(t)-(p^*+\Delta_i)||^2\Big(\frac{R_c}{R_i^*(t)}\Big) \\ \qquad\qquad : p_i(t)\in\mathcal{B}(p^*,R_c) \\ R_c k_{att}||p_i(t)-p^*|| - \frac{1}{2}k_{att}R_c R_i^*(t) \\ \qquad\qquad : p_i(t)\notin\mathcal{B}(p^*,R_c) \end{cases} \tag{8}$$

where $k_{att} > 0$ is an adjustable intensity parameter of the attractive potential field. Furthermore, $R_i^*(t) \in \mathbb{R}_{>0}$ is a normalizing distance introduced to ensure convergence of an agent at the target position after entering the target range $\mathcal{B}(p^*, R_c)$. As shown in Fig. 2, it is defined as $R_i^*(t) = ||p^* - Q_i(t)||$, where $Q_i(t)$ is the intersection with the circumscribed circle of $\mathcal{B}(p^*, R_c)$ and the directed line segment $\overrightarrow{p^* p_i(t)}$. The gradient of $U_{i,att}(p_i(t))$ follows

$$\nabla U_{i,att}\big(p_i(t)\big) = \begin{cases} k_{att}\big(p_i(t)-(p^*+\Delta_i)\big)\Big(\frac{R_c}{R_i^*(t)}\Big) & : p_i(t)\in\mathcal{B}(p^*,R_c) \\ \frac{R_c k_{att}\big(p_i(t)-p^*\big)}{||p_i(t)-p^*||} & : p_i(t)\notin\mathcal{B}(p^*,R_c) \end{cases} \tag{9}$$

As it can be noticed, when $p_i(t)$ is outside the target range $\mathcal{B}(p^*, R_c)$, $\nabla U_{i,att}$ introduces a constant attractive control that drives $p_i(t)$ towards p^*. In case that $p_i(t)$ is inside the circle, it holds that $\nabla U_{i,att}\big(p_i(t)\big) = 0$ when $||p_i(t) - \big(p^* + \Delta_i\big)|| = 0$. Furthermore, it can be verified that $U_{i,att}\big(p_i(t)\big)$ and $\nabla U_{i,att}\big(p_i(t)\big)$ are continuous at the switching boundary, i.e., the two segments of the piece-wise function are identical when $||p_i(t) - p^*|| = R_i^*(t)$.

Next, the two repulsive potential functions are designed as follows to avoid collisions against other agents and obstacles during the mission

$$U^a_{i,rep}(p_i(t)) = \begin{cases} \sum\limits_{j\in\mathbb{N}_i(t)}\Big(\frac{\Psi}{2(d_{ij,\min}-d_{ij,\max})}d_{ij}^2(t) - \frac{\Psi d_{ij,\max}}{d_{ij,\min}-d_{ij,\max}}d_{ij}(t) \\ \qquad + \frac{\Psi d_{ij,\max}^2}{2(d_{ij,\min}-d_{ij,\max})}\Big) & : d_{ij}(t)\le d_{ij,\max} \\ 0 & : d_{ij}(t) > d_{ij,\max} \end{cases} \tag{10}$$

where $d_{ij}(t) = ||p_i(t) - p_j(t)||$ denotes the distance between agent i and agent j. Ψ is the pre-set threshold for the repulsive field and $d_{ij,\max} = r_i + r_j + d_{\max}$ with $d_{\max}$ the designed maximum distance interval to activate the repulsive field.

$$\nabla U^a_{i,rep}(p_i(t)) = \begin{cases} \sum\limits_{j\in\mathbb{N}_i(t)}\Big(\frac{\Psi}{d_{ij,\min}-d_{ij,\max}}d_{ij}(t) - \frac{\Psi d_{ij,\max}}{d_{ij,\min}-d_{ij,\max}}\Big)\nabla d_{ij}(t) \\ \qquad\qquad : d_{ij}(t)\le d_{ij,\max} \\ 0 \qquad\qquad : d_{ij}(t) > d_{ij,\max} \end{cases} \tag{11}$$

where $\nabla d_{ij}(t) = \frac{p_i(t)-p_j(t)}{||p_i(t)-p_j(t)||}$. The repulsive potential function (10) is continuous and linearly dependent on the interval between agents. The motivation is to circumvent unbounded gradient as with the existing APF algorithms where the repulsive force is proportional to the reciprocal of the interval. By analogy, $\nabla U^o_{i,rep}\big(p_i(t)\big)$ with respect to obstacles can be constructed following the same form of (11), but replacing the index $j \in \mathbb{N}_i$ with $l \in \mathbb{O}_i(t)$. It is worth noting that $\mathbb{O}_i(t)$ can be shared with all neighboring agents to avoid mutual occlusion. As obstacles are not available at the beginning, generation of the trajectory $p_i(t)$ by reproducing $u_i(t)$ using the proposed APF algorithm will be necessary during the mission.

Furthermore, the maximum magnitude of the repulsive control is achieved when $d_{ij}(t) = d_{ij,\min}$, which yields $\max\{||\nabla U_{i,rep}(p_i(t))||\} = ||\Psi||$ in view of (11). Combining with the bounded attractive input (9), the resulting control law (7) can be bounded by design, which greatly facilitates tracking of the generated trajectories [29]. On the other hand, $d_{ij,\max}$ is the maximum distance to activate the repulsive field (10), therefore, agent i only connects with agents with $d_{ij}(t) \le d_{ij,\max}$. As illustrated in Fig. 3, there is no connection between agent 2 and agent 4 due to the distance $d_{24}(t)$ being longer than $d_{24,\max}$. At the next time instant, positions of agents shift, resulting in an evolution in the topology change which is shown in the figure.

Remark 1: To prevent agents from falling into a deadlock, a small perturbation can be introduced to the APF control law (7) to enforce agents escaping from deadlock [30].

IV. Theoretical Analysis

To ensure the effectiveness of the proposed algorithm, this section provides a mathematical analysis to show that all agents can be driven toward the target trajectory by the

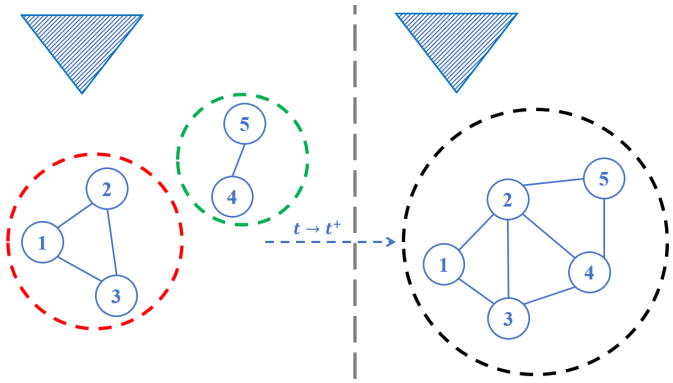


Fig. 3. Schematic diagram about topology changes of the connected agents.

proposed APF-based algorithm. The following assumption is needed to proceed with the analysis.

Assumption 4.1: The individual target formation $p_i^* + \Delta_i$ is set such that the condition $||\Delta_i - \Delta_j|| \geq d_{ij,\max}, \forall i, j \in \mathbb{A}$ is satisfied.

Assumption 4.2: There exists a time T, such that for all $t > T$, $||p_i^*(t) - p_l^o(t)|| \geq d_{il,\max}, \forall i \in \mathbb{A}, \forall l \in \mathbb{O}$.

The purpose of Assumption 4.1 is to ensure that the repulsive potential function is inactive when all agents form the desired formation. This can be achieved by a proper design of $d_{\max}$. Assumption 4.2 implies that obstacles are separated from target range $\mathcal{B}(p^*, R_c)$ in steady state so that targets can be reached without activating the repulsive potential function imposed by obstacles.

Theorem 4.1: Under Assumptions 2.1, 4.1 and 4.2, given the agent kinematic model (3) and the APF-based control law (7), agents can form the pre-defined desired formation, $p_i(t) \to p^* + \Delta_i$, $\forall i \in \mathbb{A}$ when $t \to \infty$.

Proof: Define the tracking error of an agent i

$$e_i(t) = p_i(t) - (p^* + \Delta_i), \ \forall i \in \mathbb{A} \tag{12}$$

By applying (12) to (8)-(11), $U_{i,att}$, $U^a_{i,rep}$, $\nabla U_{i,att}$ and $\nabla U^a_{i,rep}$ can be represented as functions of $e_i(t)$ rather than $p_i(t)$. Then, in view of (7), it can be shown that

$$\begin{aligned} \dot{e}_i(t) &= \dot{p}_i(t) - (\dot{p}^* + \dot{\Delta}_i) \\ &= \nabla U_{i,att}\big(p_i(t)\big) + \nabla U^a_{i,rep}\big(p_i(t)\big). \end{aligned} \tag{13}$$

It is noted that the repulsive potential functions introduced by the obstacles are ignored according to Assumption 4.2.

Consider the Lyapunov candidate with respect to $\mathbf{e}(t) = [e_1(t)^\top, e_2(t)^\top, \ldots, e_N(t)^\top]^\top$

$$V(\mathbf{e}(t)) = \sum_{i \in \mathbb{A}} \frac{1}{2} e_i(t)^\top e_i(t) \tag{14}$$

where $V(\mathbf{e}(t))$ as shown above is always positive definite. In addition, $V(0) = 0$ as $U_{i,att} = 0$ when $e_i = 0$ and $U^a_{i,rep} = 0$ when $e_i = e_j = 0$ (due to the fact that both agents are at the target positions provided $d_{ij}(t) > d_{ij,\max}$). The derivative of V along the system trajectory is

$$\begin{aligned} \dot{V}(\mathbf{e}(t)) &= \sum_{i \in \mathbb{A}} \Big(e_i(t)^\top \dot{e}_i(t) \Big) \\ &= \sum_{i \in \mathbb{A}} e_i(t)^\top \Big(\nabla U_{i,att}\big(p_i(t)\big) + \nabla U^a_{i,rep}\big(p_i(t)\big) \Big) \end{aligned} \tag{15}$$

The summation of the repulsive field gradients equals zero, and the gradient of attractive field always opposes the direction of $e_i(t)$. Therefore, $\dot{V}(\mathbf{e}(t)) < 0$, indicating that the proposed MAS is asymptotically stable and $p_i(t) \to p^* + \Delta_i, t \to \infty, i \in \mathbb{A}$. ■

V. Simulation Results

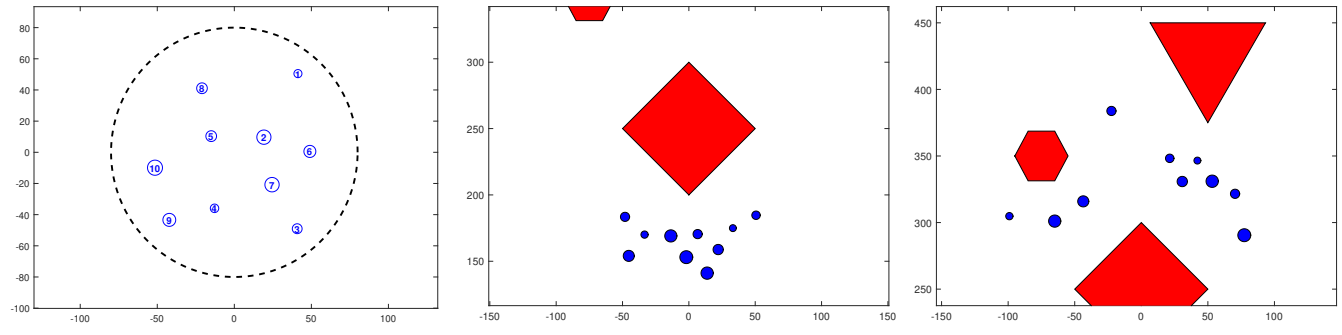
The numerical example is carried out in this section to verify the effectiveness of the proposed algorithm. Consider a MAS of ten agents, which are randomly placed within the swarm $\mathcal{B}([0,0], R_c)$ at the initial step. The final target is to achieve a desired triangular formation at the steady state. Table I summarizes the initial positions and dimensions of all agents, while the chosen parameters for swarm dimension and the parameters of the APF algorithm are provided in Table II. Simulation results are shown in Fig. 4. The agents can be driven to form the desired formation within $\mathcal{B}(p^*, R_c)$ and avoid all three obstacles in the environments during the mission by topology reconfiguration - splitting into multiple swarms and merging back when necessary.

TABLE I

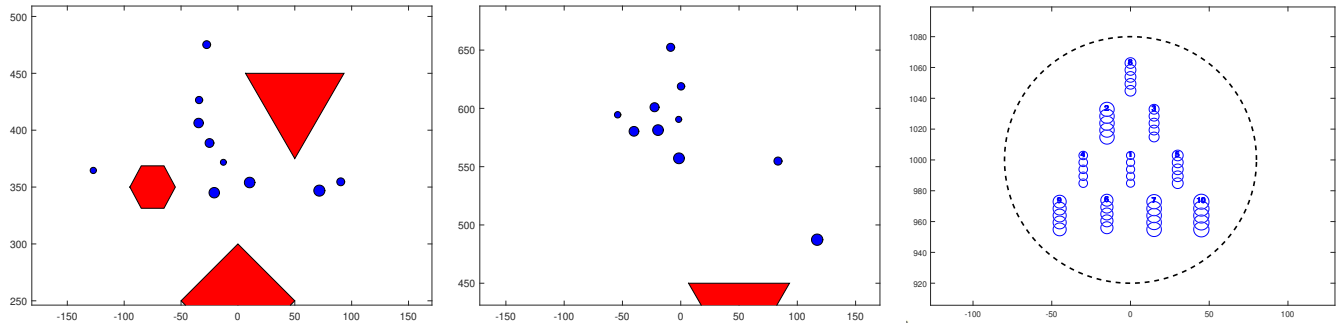
Initial Positions and Agent Radii

Agent	1	2	3	4	5	6	7	8	9	10
$p_{i,x}(0)$	41.3	19.2	40.8	-12.8	-15.0	49.0	24.5	-21.0	-42.2	-51.5
$p_{i,y}(0)$	50.5	9.7	-49.0	-35.9	10.4	0.6	-20.8	41.1	-43.4	-9.9
$r_i[m]$	2.60	4.58	3.17	2.73	3.46	3.87	4.62	3.44	4.16	4.85

Moreover, in order to highlight the advantages of the proposed algorithm, a state-of-the-art swarm-based APF control algorithm [5] is set as benchmark in the same simulation environment. In contrast to the proposed method, the method in [5] requests all agents to remain in a swarm by following a predefined virtual leader (illustrated by the red circle labeled as "v"). The comparison results are shown in Fig. 5. In details, Fig. 5(a) demonstrates how the tracking errors converge under the control of the proposed algorithm. As it can be noticed, agents form the pre-determined structure in $\mathcal{B}(p^*, R_c)$, taking 200 seconds. Meanwhile, the result of the benchmark method is illustrated in Fig. 5(b). Due to the agents being confined to a fixed area, they are unable to pass through the obstacles from both sides, resulting in a longer convergence time of 241 seconds. Additionally, as shown in Fig. 6, agent 9 and agent 10 are forced to leave the swarm to avoid collisions owing to limited space. Due to the repulsive force applied at the outer edge of the swarm by the benchmark, once an agent leaves, it cannot rejoin the swarm. Such an issue might be addressed by refining the reference, which, however, is challenging in the presence of initially unknown obstacles, and the flexibility introduced by the proposed algorithm can circumvent the challenge.

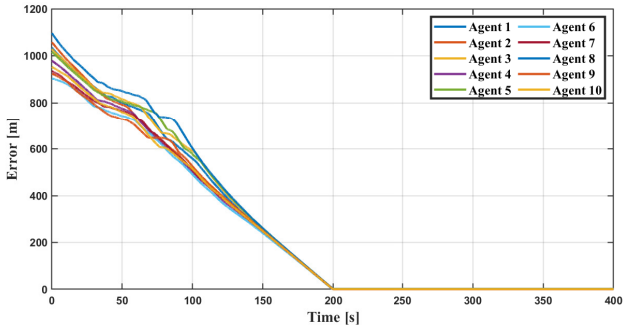


(a) $t = 0s$ (b) $t = 40s$ (c) $t = 80s$

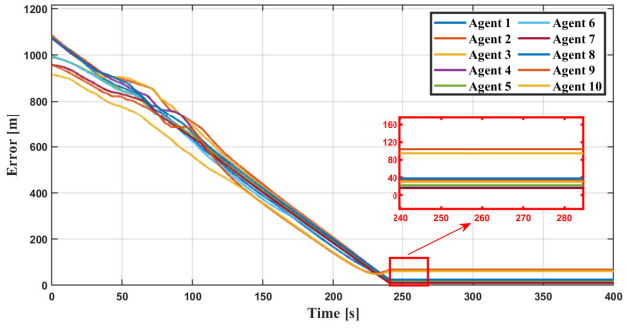


(d) $t = 92s$ (e) $t = 110s$ (f) $t = 200s$

Fig. 4. Episodes of the generated trajectories by the proposed method, where the coordinate unit is 'meter'. (a) agents' initial positions are randomly generated in the initial range $\mathcal{B}([0,0], R_c)$. (b) $t = 40$s, the diamond obstacle appears and enforces the agents to re-generate the trajectories. (c) $t = 80$s, agents pass over the first obstacle and start to tackle the hexagonal obstacle and the triangle obstacle via topology reconfiguration. (d) $t = 92$s, the proposed strategy enables all agents to make use of the small space between obstacles. (e) $t = 110$s, agents completely pass all obstacles. (f) $t = 200s$, agents reach the target swarm range $\mathcal{B}(p^*, R_c)$ and form the pre-defined formation.



(a) Tracking error $||p_i(t) - (p^* + \Delta_i)||$ of the proposed algorithm.



(b) Tracking error $||p_i(t) - p^*||$ of the benchmark.

Fig. 5. The Sub-figure (a) illustrates the tracking error $||p_i(t) - (p^* + \Delta_i)||$ of the proposed algorithm; The Sub-figure (b) shows the tracking error $||p_i(t) - p^*||$ of the benchmark.

TABLE II

PARAMETERS ABOUT APF MODEL AND SWARM SIZE

Description	Symbols	Values
Intensity parameter	k_{att}	0.08
External circle radius of swarm	R_c	$80m$
Max repulsive field gradient	Ψ	$-65m/s$
Safe distance	d_{safe}	$2m$
Max distance for repulsion activation	$d_{\max}$	$10m$

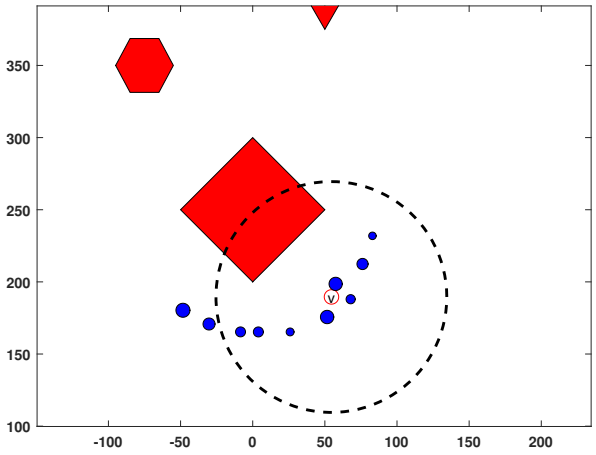


Fig. 6. Path planning following a fixed swarm configuration subject to the given virtual leader denoted by 'v'. Agents are pushed out of the swarm in event of a large obstacle.

VI. Conclusions And Future Works

This paper investigates a two-dimensional space coordination problem for multiple agents in a constrained environment. A reconfigurable distributed path planning approach is designed using the concept of the artificial potential field. Instead of confining all agents within a swarm, the proposed algorithm allows the swarm of agents to split into multiple groups or merge into a single swarm to cope with the emerging obstacles for enhanced flexibility during the task. The convergence of the algorithm is proved by using the Lyapunov theory. Numerical results verify the effectiveness of the proposed method and the benefit of enabling topology reconfiguration.

Future work consists in developing the trajectory tracking algorithm, where uncertainties can be addressed. In addition, different communication topologies will be studied to reduce the communication load during the mission.

References

[1] J. H. Holland, *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT press, 1992.

[2] Y. Zheng, S. E. Li, K. Li, F. Borrelli, and J. K. Hedrick, "Distributed model predictive control for heterogeneous vehicle platoons under unidirectional topologies," *IEEE Transactions on Control Systems Technology*, vol. 25, no. 3, pp. 899–910, 2017.

[3] H. Sun, L. Dai, and B. Chen, "Tube-based distributed model predictive control for heterogeneous vehicle platoons via convex optimization," in *2022 IEEE 25th International Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 2022, pp. 1122–1127.

[4] S. Xie, J. Hu, Z. Ding, and F. Arvin, "Cooperative adaptive cruise control for connected autonomous vehicles using spring damping energy model," *IEEE Transactions on Vehicular Technology*, vol. 72, no. 3, pp. 2974–2987, 2023.

[5] G. Franzè, G. Fedele, A. Bono, and L. D'Alfonso, "Reference tracking for multiagent systems using model predictive control," *IEEE Transactions on Control Systems Technology*, vol. 31, no. 4, pp. 1884–1891, 2023.

[6] F. Rekabi-Bana, J. Hu, T. Krajník, and F. Arvin, "Unified robust path planning and optimal trajectory generation for efficient 3D area coverage of quadrotor UAVs," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 3, pp. 2492–2507, 2024.

[7] A. Bono, B. Chen, L. D'Alfonso, and G. Fedele, "Swarm model for path tracking with reference motion profile: a diffeomorphism-based approach," *IFAC-PapersOnLine*, vol. 56, no. 2, pp. 415–418, 2023.

[8] X. Dong, Y. Li, C. Lu, G. Hu, Q. Li, and Z. Ren, "Time-varying formation tracking for UAV swarm systems with switching directed topologies," *IEEE transactions on neural networks and learning systems*, vol. 30, no. 12, pp. 3674–3685, 2018.

[9] L. Zhou, S. Leng, Q. Liu, and Q. Wang, "Intelligent UAV swarm cooperation for multiple targets tracking," *IEEE Internet of Things Journal*, vol. 9, no. 1, pp. 743–754, 2022.

[10] B. Ichter, E. Schmerling, T.-W. E. Lee, and A. Faust, "Learned critical probabilistic roadmaps for robotic motion planning," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 9535–9541.

[11] S. Erke, D. Bin, N. Yiming, Z. Qi, X. Liang, and Z. Dawei, "An improved a-star based path planning algorithm for autonomous land vehicles," *International Journal of Advanced Robotic Systems*, vol. 17, no. 5, p. 1729881420962263, 2020.

[12] M. Luo, X. Hou, and J. Yang, "Surface optimal path planning using an extended dijkstra algorithm," *IEEE access*, vol. 8, pp. 147 827–147 838, 2020.

[13] K. N. McGuire, G. C. de Croon, and K. Tuyls, "A comparative study of bug algorithms for robot navigation," *Robotics and Autonomous Systems*, vol. 121, p. 103261, 2019.

[14] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *Proceedings. 1985 IEEE International Conference on Robotics and Automation*, vol. 2, 1985, pp. 500–505.

[15] Z. Pan, C. Zhang, Y. Xia, H. Xiong, and X. Shao, "An improved artificial potential field method for path planning and formation control of the multi-UAV systems," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 69, no. 3, pp. 1129–1133, 2022.

[16] G. Fedele, L. D'Alfonso, A. Bono, and V. Gazi, "Swarm trajectories generation for target capturing with uncertain information," *IEEE Transactions on Control of Network Systems*, pp. 1–11, 2023.

[17] S. Xie, J. Hu, P. Bhowmick, Z. Ding, and F. Arvin, "Distributed motion planning for safe autonomous vehicle overtaking via artificial potential field," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 11, pp. 21 531–21 547, 2022.

[18] R. Olfati-Saber, "Flocking for multi-agent dynamic systems: algorithms and theory," *IEEE Transactions on Automatic Control*, vol. 51, no. 3, pp. 401–420, 2006.

[19] W. Bai, T. Li, Y. Long, and C. L. P. Chen, "Event-triggered multigradient recursive reinforcement learning tracking control for multiagent systems," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 1, pp. 366–379, 2023.

[20] S. G. Anavatti, S. L. Francis, and M. Garratt, "Path-planning modules for autonomous vehicles: Current status and challenges," in *2015 International Conference on Advanced Mechatronics, Intelligent Manufacture, and Industrial Automation (ICAMIMIA)*. IEEE, 2015, pp. 205–214.

[21] B. Li, H. Liu, D. Xiao, G. Yu, and Y. Zhang, "Centralized and optimal motion planning for large-scale agv systems: A generic approach," *Advances in Engineering Software*, vol. 106, pp. 33–46, 2017.

[22] C. M. Clark, "Probabilistic road map sampling strategies for multi-robot motion planning," *Robotics and Autonomous Systems*, vol. 53, no. 3-4, pp. 244–264, 2005.

[23] H. Andreasson, A. Bouguerra, M. Cirillo, D. N. Dimitrov, D. Driankov, L. Karlsson, A. J. Lilienthal, F. Pecora, J. P. Saarinen, A. Sherikov *et al.*, "Autonomous transport vehicles: Where we are and what is missing," *IEEE Robotics & Automation Magazine*, vol. 22, no. 1, pp. 64–75, 2015.

[24] X. Pan, B. Chen, S. Timotheou, and S. A. Evangelou, "A convex optimal control framework for autonomous vehicle intersection crossing," *IEEE Transactions on Intelligent Transportation Systems*, vol. 24, no. 1, pp. 163–177, 2023.

[25] K. Wu, J. Hu, Z. Li, Z. Ding, and F. Arvin, "Distributed collision-free bearing coordination of multi-UAV systems with actuator faults and time delays," *IEEE Transactions on Intelligent Transportation Systems*, 2024.

[26] D. Roy, A. Chowdhury, M. Maitra, and S. Bhattacharya, "Geometric region-based swarm robotics path planning in an unknown occluded environment," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 7, pp. 6053–6063, 2021.

[27] A. Bono, G. Fedele, and G. Franzè, "A swarm-based distributed model predictive control scheme for autonomous vehicle formations in uncertain environments," *IEEE Transactions on Cybernetics*, vol. 52, no. 9, pp. 8876–8886, 2022.

[28] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, "Robotics: modelling, planning and control," *New York, NY, USA: Springer*, pp. 415–418, 2010.

[29] S. Zhao, D. V. Dimarogonas, Z. Sun, and D. Bauso, "A general approach to coordination control of mobile agents with motion constraints," *IEEE Transactions on Automatic Control*, vol. 63, no. 5, pp. 1509–1516, 2018.

[30] J. Barraquand and J.-C. Latombe, "A monte-carlo algorithm for path planning with many degrees of freedom," in *Proceedings., IEEE International Conference on Robotics and Automation*. IEEE, 1990, pp. 1712–1717.