VIETNAM NATIONAL UNIVERSITY
UNIVERSITY OF ENGINEERING AND TECHNOLOGY

BUI DUY NAM

DISTRIBUTED CONTROL STRATEGIES FOR CHANGING MULTIPLE UAV FORMATION

MASTER THESIS
MAJOR: ELECTRONICS ENGINEERING

HANOI – 2024

VIETNAM NATIONAL UNIVERSITY
UNIVERSITY OF ENGINEERING AND TECHNOLOGY

BUI DUY NAM

DISTRIBUTED CONTROL STRATEGIES FOR CHANGING MULTIPLE UAV FORMATION

MASTER THESIS
MAJOR: ELECTRONICS ENGINEERING

Advisor: **Dr. Pham Duy Hung**
Co-Advisor: **Dr. Phung Manh Duong**

HANOI – 2024

Authorship

"I hereby declare that the work entitled "Distributed Control strategies for Changing Multiple UAV Formation" contained in this thesis is of my own and has not been previously submitted for a degree or diploma at this or any other higher education institution. To the best of my knowledge and belief, the thesis contains no materials previously published or written by another person except where due reference or acknowledgement is made."

Date: October 25, 2024

Signature: .....

Approval of Supervisors

"I hereby approve that the thesis in its current form is ready for committee examination as a requirement for the Master of Electronics Engineering at the VNU University of Engineering and Technology."

Date: October 25, 2024

Advisor: **Dr. Pham Duy Hung**

Signature: .....

Co-Advisor: **Dr. Phung Manh Duong**

Signature: .....

Abstract

Unmanned Aerial Vehicles (UAVs), commonly referred to as drones, have the potential to significantly influence various applications, including inspection, post-disaster assessment, and Search and Rescue (SAR). This is due to their exceptional agility, allowing for free movement in 3D space, and their progressively decreasing cost. To automate the aforementioned tasks using UAVs, researchers have concentrated on enhancing the capability of these vehicles to autonomously navigate unfamiliar environments, utilizing onboard sensors for pose estimation, mapping, and path planning. Advanced methods for deploying multiple robots, including formation as a popular coordination strategy, have been proposed to increase the efficiency of robotic missions, which is particularly crucial in time-sensitive applications like rescue operations. However, coordinating multiple UAVs in a formation within a constrained environment presents several challenges, including maintaining formation and avoiding collisions. Motivated by these challenges, this thesis addresses the deformation control problem of a multi-robot formation to safely navigate through narrow environments.

Aiming to the widest formation configuration, i.e. V-shape formation, and the typical assumptions related to communication and navigation, the first approach proposed a distributed self-reconfiguration control strategy. The objective is the safe navigation of a V-shape formation through a narrow space. However, although proposing a complete solution, this approach is still limited to only V-shape configuration. Addressing this limitation, in a follow-up approach, an event-based deformation control for multiple formation configurations is proposed. This method provides an effective strategy to transform the formation shape in confined spaces, by collecting data from local sensors equipped on each robot. Moreover, the strategy is demonstrated as stable via Lyapunov theory. Additionally, this type of control mission includes minimizing energy, as well as formation errors during the movement, subject to the constraints and limitations of the robot system. Therefore, an optimal control strategy is developed in the third approach, which transforms the ideal given in the previous work to the optimal solution that meets further requirements in system constraints, as well as enhances the smoothness of the movement.

With the focus on multi-robot coordination and perception-aware active planning for UAVs, the approaches and systems presented in this thesis contribute towards autonomous aerial navigation deployable in confined space scenarios. Furthermore, it is demonstrated that the use of the proposed methods is extremely beneficial for multi-robot control purposes during flights. This leads to more robust methods, contributing to the way towards more safe autonomous navigation of robotic agents.

Acknowledgements

This master's thesis would not have been possible without the help, support, and contributions from numerous people. To begin with, I would like to express my sincere gratitude to Dr. Pham Duy Hung, lecturer of the Automatic control and Robotics Lab at VNU University of Engineering and Technology, who gave me a rare opportunity to conduct my master's studies within the lab. Thank you for your enthusiastic guidance during my program, your trust in my abilities, and for inspiring me. His patient yet strict guidance helped me broaden my horizons significantly. His enthusiasm, endless passion, and professional attitude towards science and research also emerged as a considerable source of motivation for me to overcome all difficulties in research and academic activities. I am also really grateful to Dr. Phung Manh Duong for giving me a hand in finding my true passion for my academic career. His contributions, advice, and encouragement helped me to enhance my knowledge in leaps and bounds and refine my ideas during the research process and completion of my thesis. His enthusiastic guidance helped me firmly step forward in the first steps of my research career.

In addition, I would also like to thank members of the Automatic control and Robotics Lab for the time we spent together, and for the invaluable inspiration and help with numerous experiments and publications. I also want to express my appreciation to all members, colleagues, teachers, and friends from the Faculty of Electronics and Telecommunication for all of your love and support. Finally, I would like to thank all the people who reviewed this thesis and for their honest and useful feedback.

And most importantly, I would like to express my deepest gratitude to my beloved family and friends, who were always there when I needed them, especially my mother, and my father, for their unconditional love, and for always having supported me in pursuing my plans, and forever the most peaceful place for me to lean on.

Bui Duy Nam

Financial Support

The research leading to the publications and results presented in this thesis was supported by the Master, PhD Scholarship Programme of the Vingroup Innovation Foundation (VINIF), codes VINIF.2022.Ths.057 and VINIF.2023.Ths.088.

Contents

Chapter 1

Introduction

1.1 Motivation

The significant potential of unmanned aerial vehicles (UAVs) to assist humans in various tasks, such as inspecting infrastructure in dangerous areas and assessing damage after natural disasters, has been a major impetus for research into the automation of UAV missions over the past few decades [1]. Despite their many advantages, single UAV systems still face significant limitations, particularly in payload capacity, battery life, and coverage range. These limitations have driven the development of multi-UAV systems, where multiple drones cooperate to perform complex missions more efficiently, with less time and performance than a single UAV. Multi-UAV cooperation offers scalability, redundancy, and improved mission speed, making them a powerful solution for many challenging applications [2], [3].

An important aspect of multi-UAV cooperation is formation control [4], in which a group of UAVs are coordinated to fly in a specific geometric pattern. Formation control is essential to maintain spatial relationships between UAVs, ensure collision avoidance, and optimize mission performance. However, when maneuvering in environments with obstacles and limited space, maintaining an effective formation is a challenge that can fail to maintain an effective formation and a high risk of collision, especially when UAVs have to maneuver through narrow passages or confined spaces. Maintaining a desired formation in such environments requires sophisticated control strategies to dynamically adjust the formation, preventing collisions while still achieving the mission objective [5], [6].

The goal of deformation control in multi-UAV systems is to enable UAV formations to dynamically adjust their shapes to meet environmental constraints and mission requirements [4], [5]. This capability is critical to ensuring that UAVs can safely navigate narrow passageways, avoid obstacles, and maintain operational efficiency. Requirements for effective deformation control include accurate real-time communication between UAVs, robust algorithms for making decisions and performing shape reconfiguration, and reliable sensing mechanisms to detect and respond to environmental changes [7]–[14].

However, achieving effective deformation control poses significant challenges. Real-time coordination is required to ensure that all UAVs in the formation can communicate and accurately synchronize their movements to maintain the desired shape. Developing robust obstacle avoidance algorithms is critical, as the formation must be able to detect and maneuver around obstacles without losing cohesion. Environmental adaptability is

another key challenge, requiring control strategies that can dynamically adjust the formation in response to changing environmental conditions. Scalability is critical to ensure that these control algorithms can handle varying numbers of UAVs without compromising performance. Additionally, robustness is crucial in maintaining formation integrity despite potential issues such as communication latency, UAV malfunction, or sensor inaccuracy. Addressing these challenges is essential to ensure the reliability and effectiveness of deformation control in multi-UAV systems.

This thesis aims to address the challenges of autonomous navigation by focusing on the coordination of multiple UAVs in deformation control, i.e. changing their shape based on environmental information. The thesis focuses on studying various formation-changing strategies for a robot swarm, particularly in navigating through unknown narrow passages, with a focus on applications such as search and rescue.

1.2 Approaches and Background

Motivated by the application areas and challenges mentioned above, the research conducted during this thesis aims to promote the autonomy of UAV formations in actively performing distributed formation transformations based on information collected from the environment. We specifically focus on studying effective deformation control strategies for various types of formations. This section provides a summary of key approaches followed in this thesis.

One of the common formation control methods is to use centralized formation control [4], [15], which involves a single control or command unit coordinating the movements and actions of all UAVs in the formation. In this method, a central controller collects information from all UAVs, processes the information to determine the optimal formation strategy, and sends commands to each UAV to perform the required operations. This method simplifies the coordination process because it reduces the problem to a single decision point, ensuring that all UAVs operate in sync based on the instructions of the central controller.

The main advantage of centralized formation control [4], [15]–[17] is the ability to achieve precise and coordinated movements across the entire formation. Since the central controller has a comprehensive view of the system, it can optimize the formation for various parameters such as fuel efficiency, coverage, and collision avoidance. In addition, this approach facilitates the implementation of complex maneuvers and formation changes, as the central controller can effectively manage the overall strategy.

However, centralized formation control also has significant disadvantages [4], [17]–[19]. The central controller becomes a single point of failure; if it malfunctions or loses contact with the UAV, the entire formation may collapse, i.e. the robustness is not guaranteed. Furthermore, the computational and communication costs on the central controller can be significant, especially in large-scale formations, leading to potential delays and bottlenecks, i.e. the scalability is limited. These limitations can hinder the scalability and robustness of the system, especially in dynamic and unpredictable environments.

In contrast, distributed formation control [4], [18] is often used because it relies on decentralized decision-making, where each UAV acts based on local information and in-

teracts with nearby UAVs. Instead of a single controller, the formation is maintained through a set of local rules and behaviors that each UAV follows. This approach mimics natural systems, such as flocks of birds or schools of fish, where complex group behaviors emerge from simple individual actions.

Distributed formation control offers several advantages, particularly in terms of scalability and robustness [4], [11], [17], [18], [20]. Since each UAV makes decisions independently, the system can be easily expanded to accommodate more UAVs without overloading the central controller. This decentralization also enhances the fault tolerance of the system; if one UAV fails, the rest can continue to operate and adapt to changing circumstances. Furthermore, distributed control allows for better flexibility and adaptability in dynamic environments, as each UAV can react to local changes and maintain formation without relying on centralized commands. Therefore, in this thesis, we focus on designing distributed controllers for their scalability and robustness.

In search and rescue (SAR) operations, the V-formation is frequently used because this shape optimizes coverage by maintaining an even distribution of UAVs, ensuring a comprehensive search without gaps [21]–[23]. The improved line-of-sight communication in the V-formation facilitates real-time data sharing and coordination, which is critical for timely decision-making in SAR missions. Additionally, the flexibility and adaptability of the V-formation allows UAVs to move effectively across diverse and challenging terrains, making it a valuable strategy for locating survivors and assessing disaster areas quickly and efficiently. There are many studies [21], [24], [25] providing insights into how to effectively maintain a V-formation in the environment. However, studies regarding the V-formation's ability to maneuver through confined environments are quite limited.

To address these limitations in the context of V-shape formation movement through a constrained environment, in *Method I*, we propose an efficient behavioral strategy capable of automatically observing and self-coordinating to maintain the formation and appropriately expand the formation according to the shape of the narrow space, with the goal of efficiently and safely navigating the robot formation.

In practice, various formation shapes offer more potential applications than the V-formation, highlighting the need for a more general formation control method. Consequently, *Method II* extends the formation transformation algorithm to accommodate different formation shapes, demonstrating its stability through Lyapunov stability theory.

The two methods above focus exclusively on primitive motion and do not address the constraints and limitations of the system. Their method relies on the potential field approach influenced by numerous parameters, potentially complicating the control process. Therefore, *Method III* adopts the model predictive control (MPC) method, which not only provides optimal control signals but also effectively manages the system's constraints and limitations. This approach ensures more precise and reliable formation control, enhancing overall performance and applicability in various operational scenarios.

1.3 Contributions

This section details the core contributions of the research carried out in this master's thesis, including a complete list of publications and open-source software libraries derived

from the methods contributing to this thesis. Next, we provide a list of all student projects supervised during the master's studies.

1.3.1 Research Contributions

The general objective of this thesis was to design control strategies to guide the formation of multiple robots through confined spaces, especially in narrow environments. These proposed approaches were presented as follows:

Method 1. Self-Reconfigurable V-Shape Formation of Multiple UAVs in Narrow Space Environments

As motivated by Section 1.1, multi-robot systems have the mission to form a V-shape and navigate through a narrow and confined space. Aiming to develop an effective control strategy in the confined space, this work proposed a self-reconfigurable V-shape formation control algorithm for multiple UAVs operating in a narrow space where the formation can be formed and maintained the desired V-shape from a random initial position and during the movement. Moreover, the formation can autonomously reconfigure its V-shape by expanding/shrinking its two V-wings to avoid collisions with obstacles and maintain safe distances among the UAVs. The primary contribution of this work is the design of the self-reconfigurable control strategy for distributed multi-UAV systems cooperating in V-shape formation. Moreover, the behaviors are presented to contribute to the distributed strategy that UAV systems can safely avoid obstacles and effectively maintain their V-shape.

Method 2. Event-based Deformation Control Strategy for Time-varying Formation in Confined Space

Multi-robot formations require numerous different shapes to adapt to various environments. Aiming to expand the wide application given in *Method 1* to multiple types of formations in confined space environments, this work proposed an event-based deformation control strategy based on the artificial potential field (APF) which enhances the safety of multi-robot time-varying formation (TVF) in a confined space. In a collision-free environment, i.e. without any obstacles, the proposed approach ensures that multi-robot systems maintain their desired configuration and inter-agent collision-free. Additionally, the local sensor equipped in each robot can detect the confined space, which contributes to the configuration change if needed by applying the scale, rotation, or transformation into the straight line topology to ensure collision-free flight.

Method 3. MPPDC: Model Prediction-based Perceptual Deformation Control for Multiple Robots in Narrow Space Environments

Aiming to improve the smoothness and optimize the motion control of multi-UAV formation given in *Method 2*, an optimal deformation control strategy is presented for a decentralized multi-robot team to ensure safety in narrow space environments. The core contribution is in a perceptual deformation control strategy to navigate a multi-robot formation moving through narrow environments effectively. Each robot is equipped with local sensors and communication modules to collect information from the surrounding environment and its neighbors for distributed decision-making. The formation thus can be shrunk/expanded or transformed to the line formation according to the environment. Particularly, a proposed strategy is formulated from a model prediction-based control strategy to achieve the navigation requirements of maintaining formation, velocity, and

direction, while effectively avoiding collisions.

1.3.2 List of Publications

During the master's studies, the following publications were achieved, with invaluable contributions from the co-authors. Furthermore, the author had the opportunity to present some of these works at international conferences. We list the publications in chronological order.

Publications included in this Thesis

- **Duy-Nam Bui**, Manh Duong Phung and Hung Pham Duy, "Self-Reconfigurable V-Shape Formation of Multiple UAVs in Narrow Space Environments," *2024 IEEE/SICE International Symposium on System Integration (SII)*, Ha Long, Vietnam, pp. 1006–1011, 2024.
- **Duy-Nam Bui**, Manh Duong Phung, Trung-Dung Ngo and Hung Pham Duy. "Event-based Deformation Control Strategy for Time-varying Robot Formation in Confined Space," in *Preprint*, 2024.
- **Duy-Nam Bui**, Manh Duong Phung, and Hung Pham Duy. "MPPDC: Model Prediction-based Perceptual Deformation Control for Multiple Robots in Narrow Space Environments," in *Preprint*, 2024.

Other publications

- **Duy-Nam Bui**, Thuy Ngan Duong and Manh Duong Phung, "Ant Colony Optimization for Cooperative Inspection Path Planning Using Multiple Unmanned Aerial Vehicles," *2024 IEEE/SICE International Symposium on System Integration (SII)*, Ha Long, Vietnam, pp. 675–680, 2024.
- **Duy-Nam Bui** and Manh Duong Phung, "Radial Basis Function Neural Networks for Formation Control of Unmanned Aerial Vehicles," in *Robotica*, vol. 42, pp. 1842–1860, 2024.
- **Duy-Nam Bui**, Thu Hang Khuat, Manh Duong Phung, Thuan-Hoang Tran, Dong LT Tran, "Optimal Motion Planning for Unmanned Aerial Vehicles in Unknown Environments," *2024 International Conference on Control, Robotics and Informatics (ICCRI)*, Da Nang, Vietnam, 2024.
- Thi Thuy Ngan Duong, **Duy-Nam Bui**, and Manh Duong Phung, "Navigation Variable-based Multi-objective Particle Swarm Optimization for UAV Path Planning with Kinematic Constraints," in *Neural Computing and Applications*, 2024.

1.3.3 Open-source Software

Some of the works developed within this thesis have been released publicly for free use by the research community. Namely, these are:

- The (multiple) UAVs simulator on Gazebo software in the loop https://github.com/duynamrcv/hummingbird_simulator.
- The nonlinear model predictive control (MPC) for UAVs low-level controller https://github.com/duynamrcv/hummingbird_nmpc.
- The self-reconfiguration strategy for V-shape formation of multiple UAVs: https://github.com/duynamrcv/reconfigurable_vshape.
- The distributed formation changing strategy for multiple UAVs: https://github.com/duynamrcv/edc.
- The model prediction-based perceptual deformation control for multiple UAVs: https://github.com/duynamrcv/mppdc.

1.3.4 List of Supervised Students

During this master studies, the author had the opportunity to (co)-supervise the following Bachelor students at VNU University of Engineering and Technology.

Bachelor Theses and Studies on Robotics Engineering, and Electronics and Communications Engineering Technology

- Khuat, Thi Thu Hang (Spring 2023): "Polar coordinate-based differential evolution for moving target search using vision sensor on unmanned aerial vehicles".
- Nguyen, Trung Hieu (Spring 2024): "Distributed model predictive control for multi-UAVs in obstacle environments".

Chapter 2

Self-reconfigurable V-shape Formation of Multiple UAVs in Narrow Space Environments

Published in:
The 2024 IEEE/SICE International Symposium on System Integration (SII 2024), 2024
DOI: 10.1109/SII58957.2024.10417519

Abstract

This chapter presents the design and implementation of a self-reconfigurable V-shape formation controller for multiple unmanned aerial vehicles (UAVs) navigating through narrow spaces in a dense obstacle environment. The selection of the V-shape formation is motivated by its maneuverability and visibility advantages. The main objective is to develop an effective formation control strategy that allows UAVs to autonomously adjust their positions to form the desired formation while navigating through obstacles. To achieve this, we propose a distributed behavior-based control algorithm that combines the behaviors designed for individual UAVs so that they together navigate the UAVs to their desired positions. The reconfiguration process is automatic, utilizing individual UAV sensing within the formation, allowing for dynamic adaptations such as opening/closing wings or merging into a straight line. Simulation results show that the self-reconfigurable V-shape formation offers adaptability and effectiveness for UAV formations in complex operational scenarios.

Keywords: unmanned aerial vehicles, multi-robot system, distributed control, formation control, reconfiguration

2.1 Introduction

Unmanned aerial vehicles (UAVs) has gained significant attention in recent years due to their potential applications in various fields, including surveillance, search and rescue operations, and infrastructure inspection [26], [27]. One crucial aspect of UAV operations is their ability to navigate and maintain formations effectively, especially in complex environments with obstacles. The formation control of multiple UAVs thus plays a vital role in achieving coordination and efficient mission execution [24], [28].

Formation control in multi-robot systems refers to the coordination and control strate-

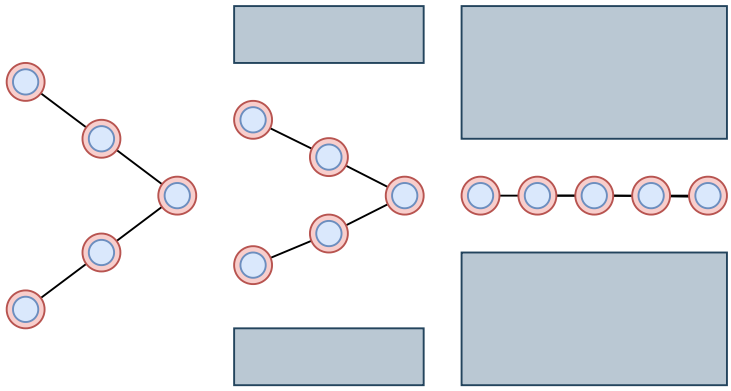


Figure 2.1: The self-reconfigurable V-shape formation can adjust its shape and navigate through narrow passages.

gies used to form, maintain, and transform formations among a group of robots [7], [29]. In scenarios with dense obstacles or narrow spaces, UAV formations need to adapt and reconfigure their shape to navigate through or around obstacles [21], [30], [31], as described in Figure 2.1. This self-reconfigurable capability enables the UAVs to overcome challenging terrain, narrow passages, or complex structures, thereby enhancing their maneuverability and overall mission success. One commonly adopted formation shape is the V-shape configuration, which offers advantages in terms of stability, visibility, and aerodynamic efficiency [21], [22]. In [21], a formation control algorithm is proposed for multi-UAV systems where the UAVs autonomously adjust their positions within a V-shape to avoid collisions with obstacles. In [23], a splitting and merging algorithm is proposed for multi-robot formations in environments presented by static and dynamic obstacles. The work in [32] presents a switching strategy of a region-based shape controller for a swarm of robots to deal with the obstacle-avoidance problem in complex environments.

Recently, advancements in path planning and obstacle avoidance techniques have contributed to the development of self-reconfigurable formation control algorithms. In [31], the angle-encoded particle swarm optimization algorithm is developed for the formation of multiple UAVs used in vision-based inspection of infrastructure. By incorporating constraints related to flight safety and visual inspection, the path and formation can be combined to provide trajectory and velocity profiles for each UAV. The work in [33] developed a novel optimization method for multi-UAV formation to achieve rapid and accurate reconfiguration under random attacks. In [34], multi-UAV reconfiguration problems are modeled as an optimal problem with task assignment and control optimization. However, the focus of their work was on maintaining a fixed formation shape rather than self-reconfiguration in the presence of obstacles.

In this work, we propose a self-reconfigurable V-shape formation control algorithm for multiple UAVs operating in narrow space where the formation cannot maintain its initial shape when moving through this space. The algorithm allows the UAVs to form, maintain and reconfigure the desired V-shape formation by expanding/shrinking its two V-wings to avoid collisions with obstacles and maintain distances among the UAVs. According to the proposed design of distributed behaviors, the V-shape formations can open/close wings or merge into a straight line. Based on it, the UAVs can navigate through narrow passages, bypass obstacles, and optimize their trajectory in accordance to environmental conditions. The main contributions of our work are twofold: (i) develop a self-reconfiguration strategy

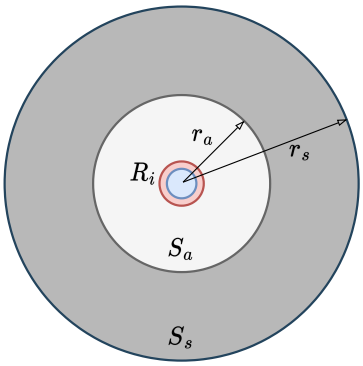


Figure 2.2: The sensing range r_s and alert range r_a ($r_a < r_s$) of a UAV R_i

that can adjust its shape in narrow space environments; and (ii) propose reconfiguration behaviors that navigate UAVs and maintain their shape.

2.2 V-Shape Formation Design

The V-shape formation is chosen due to its advantages in improving maneuverability and enhancing visibility for UAVs. Its modeling and design principles are presented as follows.

2.2.1 UAV model

The formation consists of n identical UAVs, each equipped with sensory modules for positioning and navigation such as Lidar, GPS and inertial measurement unit (IMU). The UAV is also equipped with a communication module that allows peer-to-peer communication among the UAVs. At height h, a UAV R_i is modeled as a particle moving in a 2D plane located at that height with position p_i and heading angle ψ_i. The single-integrator kinematic model of UAV R_i can be expressed as follows:

$$\dot{p}_i = u_i, \tag{2.1}$$

where $u_i = [u_{ix}, u_{iy}]^T$ is the velocity vector of UAV R_i. The heading angle ψ_i then can be obtained as:

$$\psi_i = \text{atan2}(u_{iy}, u_{ix}). \tag{2.2}$$

The communication range of each UAV R_i is divided into two areas including the sensing area S_s with radius r_s and the alert area S_a with radius $r_a < r_s$ so that $S_a \subset S_s$, as illustrated in Figure 2.2.

2.2.2 V-shape formation

In this work, the V-shape formation is constructed by two wings [21], as shown in Figure 2.3. The wings are described by the desired distances between consecutive UAVs, d, and the bearing angle between the formation heading and each wing, α. Without loss of generality, choose UAV R_l, with $l = \lceil n/2 \rceil$, as the leader UAV located at the forefront

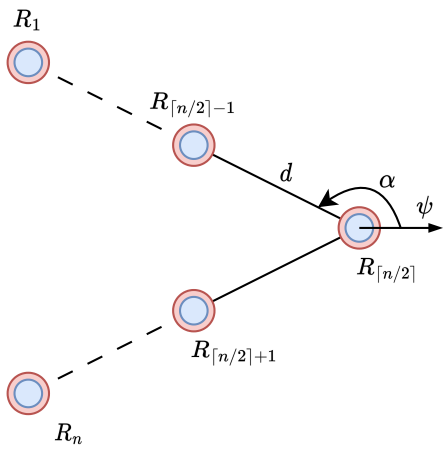


Figure 2.3: Illustration of the V-shape formation

of the formation. The desired distance d_i and angle α_i between UAV R_i, $i \neq l$, and UAV R_l are determined as follows:

$$\begin{aligned} d_i &= d\,|l - i|\,, \\ \alpha_i &= \begin{cases} \psi_l + \alpha & \text{if } i < l \\ \psi_l - \alpha & \text{if } i > l \end{cases} \end{aligned} \tag{2.3}$$

Thus, the desired position of UAV R_i can be obtained as follows:

$$p_i^d = p_l + d_i \begin{bmatrix} \cos\alpha_i \\ \sin\alpha_i \end{bmatrix}. \tag{2.4}$$

2.3 Distributed Formation Control Strategy

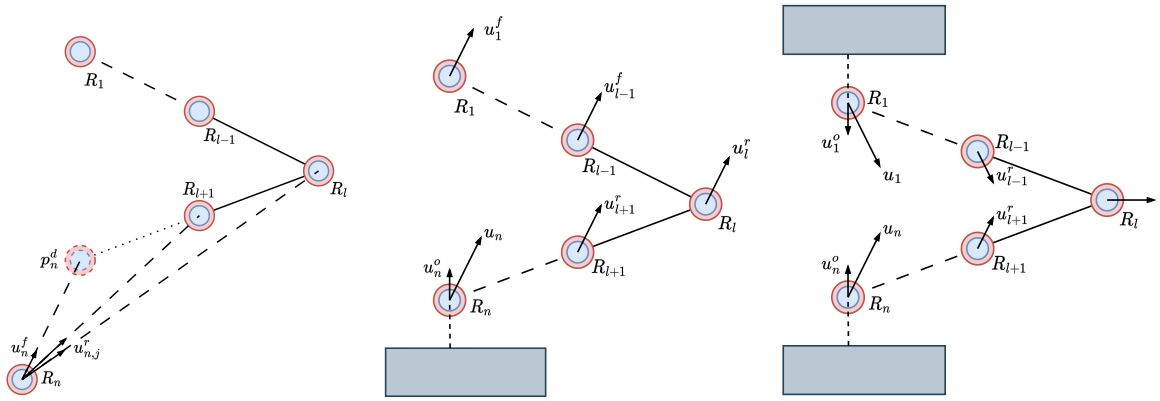
The formation control strategy aims to form, maintain and self-reconfigurate the V-shape formation in response to obstacles and narrow passages during UAV navigation. This adaptation is achieved through either expanding/shrinking two wings of the V-shape. It allows the UAV formation to navigate safely within confined spaces without encountering collisions. The proposed algorithm operates based on the use of distributed behavior-based control and artificial potential field approaches so that individual UAVs can make decisions and adjust the formation shape as needed. The strategy consists of two parts: maintaining formation and reconfigurating formation.

2.3.1 Formation maintenance strategy

Behavior-based control is the approach that combines a set of distributed control modules, called behaviors, to achieve the desired objective [7], [35]. In this work, the UAV formation is maintained based on the combination of the following behaviors.

Formation behavior

The formation behavior aims to guide UAVs to achieve their desired positions within the predefined formation. According to (2.4), the desired position p_i^d of UAV R_i in the



(a) Disruption in formation (b) Obstacles from one side (c) Obstacles from both sides

Figure 2.4: Self-reconfiguration of the V-shape formation based on the mechanism of pliers or scissors

formation can be obtained. Inspired by [21], [36], we define the formation behavior as follows:

$$u_i^f = -k_f \left(p_i - p_i^d\right) + u_l, \tag{2.5}$$

where $k_f > 0$ is a positive formation gain.

Goal reaching behavior

This behavior navigates the formation towards the desired location. To accomplish this objective, a target-tracking controller is formulated based on the relative position between the leader UAV and the goal. Let p_g be the goal position that the formation needs to reach. The goal reaching behavior is constructed as follows:

$$u_i^g = -k_g \left(p_i - p_g\right), \tag{2.6}$$

where $k_g > 0$ is a positive tracking gain.

Obstacle avoidance behavior

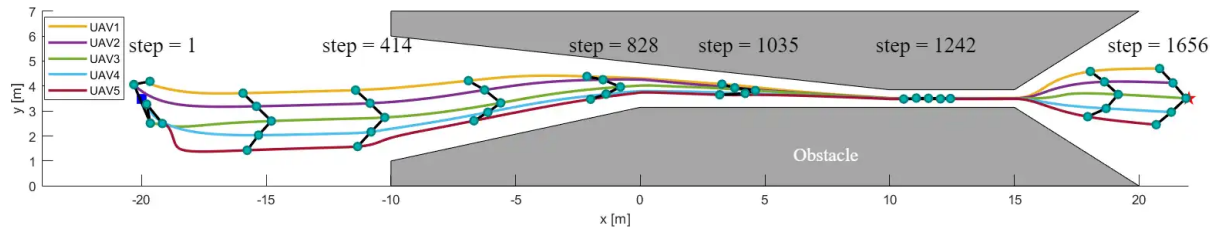
During operation, the formation must avoid obstacles present in the environment. Let p_{io_h} be the closest point on the boundary of obstacle o_h within the sensing range of UAV R_i. When that UAV senses obstacle o_h, it will create a thrust to maneuver and avoid the obstacle. The thrust is directed as follows:

$$u_{ih}^o = \begin{cases} -k_o \left(\dfrac{1}{d_{io_h}^2} - \dfrac{1}{r_s^2}\right) \dfrac{p_i - p_{io_h}}{\|p_i - p_{io_h}\|}, & \text{if } d_{io_h} < r_s \\ 0. & \text{otherwise} \end{cases} \tag{2.7}$$

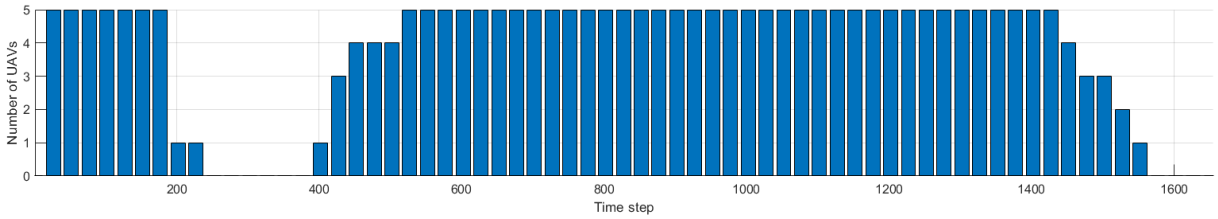
where $k_o > 0$ is a positive gain; d_{io_h} is the distance between UAV R_i and obstacle o_h. When considering all obstacles, the obstacle avoidance behavior of UAV R_i are obtained as follows:

$$u_i^o = \sum_{h=1}^{m} u_{ih}, \tag{2.8}$$

where m is number of observable obstacles within the sensing range of UAV R_i.



(a) Trajectories of the UAVs in the formation



(b) Number of UAVs activating reconfiguration behaviors over time

Figure 2.5: Simulation result of the V-shape formation moving through a narrow passage

Collision avoidance behavior

Apart from avoiding obstacles, the control algorithm also needs to adjust the UAV positions to avoid collision among them. To address this, we propose that UAVs R_i and R_j that are not in the same wing but within each other's sensing area, i.e., $\|p_{ij}\| < r_s$, will exert a repulsive force to preventing the UAVs from entering the alert area S_a. Let $p_{ij} = p_i - p_j$. The collision avoidance behavior is determined as follows:

$$u_{ij}^c = k_c \frac{e^{-\beta_c(\|p_{ij}\| - r_a)}}{\|p_{ij}\| - r_a} \frac{p_i - p_j}{\|p_i - p_j\|}, \tag{2.9}$$

where $k_c > 0$ is a positive collision gain.

2.3.2 Self-reconfigurable formation strategy

Inspired by the mechanics of pliers and scissors, which alter their shape through the application of opposing forces on their handle arms, the V-shape formation can open or close its wings based on the exertion of external forces. In our work, the force is produced based on the difference in the distances among the UAVs and their desired distances. Specifically, in case the formation encounters disruptions arising from improper positioning, as illustrated in Figure 2.4a where UAV R_n deviates from the alignment, the combined force acts to guide it toward its desired location.

In the scenario depicted by Figure 2.4b where a force is exerted from one side, UAV R_n responds by generating a thrust u_n^o to avoid a potential collision with obstacles thus resulting in the control signal u_n. Accordingly, other UAVs on the same V-wing, including the leader UAV, also adjust their position based on the behavior control signal u_i^r. As the leader UAV R_l changes its position, the UAVs on the opposing wing realign themselves by formation behavior u_i^f. As a result, the whole UAV formation tends to move towards the other side.

In another scenario, when obstacles impact the formation from both opposing sides, as depicted in Figure 2.4c, they affect UAVs R_1 and R_n, leading to the generation of obstacle

Table 2.1: Statistical evaluation of the proposed strategy for several different scenarios

Scenario	Number UAVs	d	α	Average error (m)	Min distance (m)	Average distance of consecutive UAVs (m)
1	3	1.0	$3\pi/4$	0.12333	0.48557	0.98788
2	5	1.0	$3\pi/4$	0.12068	0.37383	0.96575
3	3	0.8	$5\pi/6$	0.10942	0.48988	0.87599
4	3	1.0	$4\pi/5$	0.13666	0.47581	1.0943
5	5	0.8	$3\pi/4$	0.111	0.41279	0.88913

avoidance behaviors denoted as u_1^o and u_n^o. Other UAVs on the same wing respond by generating reconfiguration behaviors u_i^r that adjust the UAVs' position accordingly. As a result, the V-shape formation is able to shrink its wing to travel through narrow passages.

In our work, the aforementioned reconfiguration idea is implemented by the following equation:

$$u_{ij}^r = k_r \frac{|\|p_{ij}\| - d_{ij}|^{\beta_r}}{\left(\|p_{ij}\| - r_a\right)^2} \frac{p_i - p_j}{\|p_i - p_j\|}, \tag{2.10}$$

where $k_r > 0$ is a positive reconfiguration gain, $\beta_r > 0$ is the smoothness factor, d_{ij} is the desired distance between two UAVs R_i and R_j n the same wing, $d_{ij} = d\,|i - j|$. In (2.10), the term $|\|p_{ij}\| - d_{ij}|$ enables the UAVs to adjust their positions so that the desired distances among the UAVs are maintained. This behavior is also used as a collision avoidance behavior for the UAVs in the same wing.

2.3.3 Overall strategy

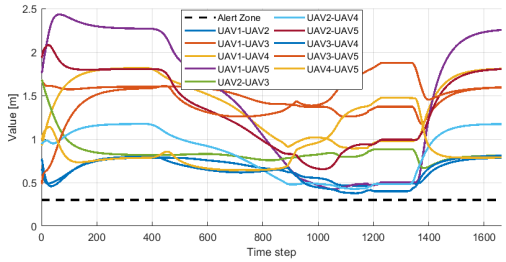
The overall distributed control strategy is obtained by combining the behaviors from all UAVs as follows:

$$u_i = \begin{cases} u_i^g + u_i^r + u_i^c + u_i^o, & \text{if assigned as leader} \\ u_i^f + u_i^r + u_i^c + u_i^o. & \text{otherwise} \end{cases} \tag{2.11}$$

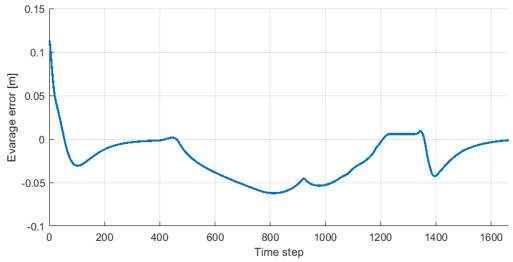
According to this function, the behaviors are automatically triggered in response to external influences or disturbances encountered by the formation. Once the desired state is achieved, the behavioral values are maintained resulting in stable operation of the UAV formation.

2.4 Results and Discussion

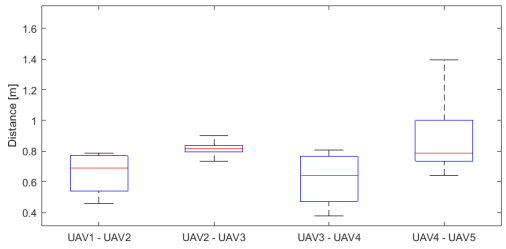
In this section, we evaluate the performance of the proposed control strategy through different simulation scenarios.



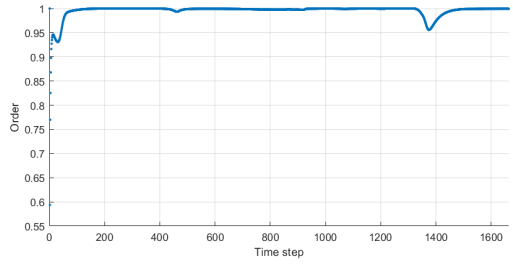
(a) Distances between each pair of UAVs over time



(b) The average distance error of UAV formation



(c) The distance between consecutive UAVs



(d) Values of the *order* metric Φ over time

Figure 2.6: Evaluation of the proposed algorithm

2.4.1 Simulation setup

In the simulation, the UAV has the alert radius $r_a = 0.3$ m and the sensing radius $r_s = 2$ m. The control period is set at 0.02 s. The maximum speed of each UAV is 2.0 m/s. The V-shape formation is defined with $d = 0.8$ m and $\alpha = 0.3\pi/4$ rad. In our evaluation, 5 UAVs with V-shape formation are operated in the area of 46 m $\times$ 7 m with two large obstacles arranged to form a narrow passage as shown in Figure 2.5.

2.4.2 Results

Figure 2.5 shows the trajectories of the UAVs moving in the environment under the guidance of the proposed distributed controller. Initially, the UAVs are randomly positioned around a starting point (step 1). They then self-adjust to form the desired V-shape formation (step 414) based on the control signals generated by the formation and reconfiguration behaviors. As they encounter obstacles, the UAVs start to adjust their formation. This includes deforming the formation (steps 414-1035) and transitioning to a straight-line formation (step 1242) to navigate through narrow gaps. Upon successfully circumventing the obstacles, the UAVs readjust to form the desired V-shape and proceed toward the goal position (step 1656). The result can be verified in the simulation video shown in footnote [1].

Figure 2.5b shows the number of UAVs activating their reconfiguration behavior over time. It can be seen that the UAVs activate this behavior when shaping the formation at the initialization stage, and during the process of adjusting their formation to adapt to the environment structure.

The statistical results of the evaluation of the proposed strategy are depicted in Figure

[1]Simulation video: https://youtu.be/_6u7yMNOySc

2.6. Figure 2.6a shows the distances between the UAVs over time. It can be seen that those distances are all greater than the alert radius, which confirms the effectiveness of the control algorithm in avoiding collision among the UAVs.

Figure 2.6b presents the average distance error of the UAV formation over time. Initially, the error is large since the UAVs have not formed the desired shape. After the control algorithm realigns the UAV to their desired positions, the error quickly converges toward zero. While the formation navigates through the narrow passage, the error remains small, less than 0.06 m. Figure 2.6c shows the average distance between consecutive UAVs. It can be seen that the average distance fluctuates around the desired value for the V-shaped formation ($d = 0.8$ m), which is desirable for the formation.

To further evaluate the performance of the proposed controller, an *order* metric Φ that measures the similarity in the UAVs' direction is used [37]. It takes the values in range $[0, 1]$ and is computed as follows:

$$\Phi = \frac{1}{n} \left\| \sum_{i=1}^{n} [\cos \psi_i, \sin \psi_i]^T \right\|. \tag{2.12}$$

According to (2.12), the order metric Φ is close to 1 when all UAVs have the same heading angle. Figure 2.6d shows the value of Φ in our simulation. It can be seen that Φ is close to 1 during the movement of the formation, even when the formation avoids obstacles or traverses through a narrow passage. Changes in the heading angle increase when exiting the passage since the UAVs in the formation need to realign to the origin shape. However, the order metric then quickly converges to 1 when the UAVs form their desired formation. The results thus confirm the validity of the proposed control algorithm.

To further evaluate the performance of the proposed method, simulations on various scenarios such as narrow passages of varying widths and dense obstacle areas have been conducted. In addition, the number of UAVs and V-shape parameters, the desired distance, d, and the desired bearing angle, α, are also varied. The results, including the average formation error, the closest distance between two UAVs, and the average distance between consecutive UAVs, are summarized in Table 2.1. It is evident that the average formation error approximates $0.1m$, which is sufficient for the formation to maneuver in narrow spaces. Furthermore, the minimum distance between any pair of UAVs is larger than the collision threshold, r_a, and thus meets the requirement for collision avoidance. Finally, the average distance between consecutive UAVs closely aligns with the desired value, indicating the stability in the formation shape. These results confirm the validity and effectiveness of our proposed control strategy.

2.5 Conclusion

In this work, we have presented a new behavior-based controller to address the problem of UAV formation in narrow space environments. We developed several behaviors for each UAV and then proposed a function to combine them. Our approach allows the UAVs to form a V-shape formation with the capability to adjust their wings to avoid obstacles and travel through narrow passages. A number of simulation evaluations have been conducted and the results show that our control strategy is not only able to navigate the UAVs to form the desired V-shape formation but also provide them with the capability

to reconfigure themselves to circumvent obstacles, avoid collisions, and traverse narrow passages in complex environments.

Chapter 3

Event-based Deformation Control Strategy for Time-varying Formation in Confined Space

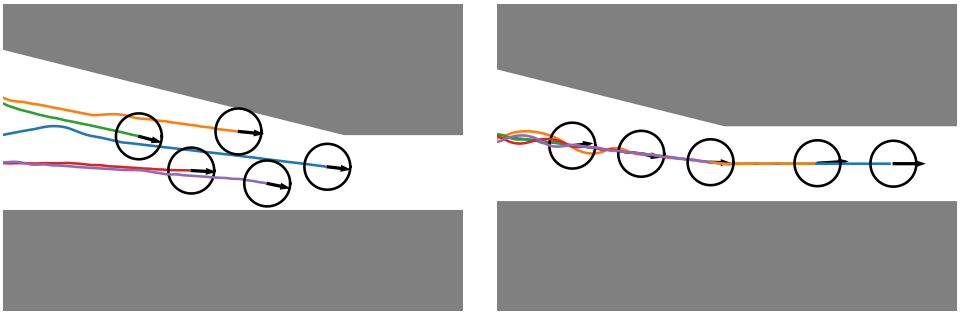
Published in:
Preprint, 2024

Abstract Formation control plays a crucial role in the coordination of multi-robot systems. Particularly in confined space environments, the navigation of a formation may risk potential collision due to space limitations. In this paper, we propose an event-based deformation control strategy based on the artificial potential field that enhances the safety of multi-robot time-varying formation (TVF) in a confined space. Without limited environments, the proposed approach ensures that multi-robot systems maintain their desired configuration and inter-agent collision-free. Furthermore, when the confined space is detected through the local sensor equipped in each robot that does not have enough space to maintain the original formation, the configuration is changed by scale, or transform into the straight line topology, which ensures collision-free flight. We verified the efficacy, correctness, and superiority of our proposed method through simulation and evaluation based on several metrics. The comparison is also conducted to highlight the effectiveness of the proposed method.

Keywords: multi-robot system, time-varying formation, event-triggering, obstacle avoidance, artificial potential fields

3.1 Introduction

With the in-depth research on networked multi-agent technology, multi-robot systems (MRSs) have rapidly developed towards autonomy, offering various applications in various areas over the last few decades [38], [39]. In multi-robot control, formation control plays a crucial key technology, in which robots are required to form and maintain a desired configuration [4], [15]. In fixed-configuration formation control, achieving the desired configuration involves setting a specific target distance for each swarm agent. However, with the increasing diversity and complexity of formation tasks, it is crucial to adjust the formation method based on specific task requirements in real-time. Consequently, time-varying formation (TVF) control has become essential for practical robot applications [40],



(a) Pure formation control (b) Proposed approach

Figure 3.1: We develop an event-based deformation control method to guide a TVF through confined spaces, including narrow gaps. *Left:* Motion path of the TVF using purely behavior-based formation control [7], [12], which are collisions with surrounding obstacles. *Right:* Motion path of the TVF using our proposed approach, which safely navigates through the narrow gap.

[41].

Research in [7], [42], [43] proposed that the motion of a biological swarm can be described by the combination of three behavioral rules, including (i) cohesion, which brings each agent closer to its neighbors, (ii) repulsion, which drives each agent away from its neighbors to avoid collisions, and (iii) alignment, which steers each agent towards the average heading of its neighbors. These rules apply to each agent simultaneously. In goal-oriented swarm flight, alignment behavior is replaced by migration behavior, guiding each UAV in a desired migration direction with a preferred speed [44]. In the obstacle environments, the navigation of a UAV swarm can be enhanced by introducing a fourth behavior, collision avoidance, which navigates UAVs around obstacles [9], [27], [45], [46]. Mathematically, these rules can be modeled using virtual forces with potential fields, which encode the desired behaviors of the swarm.

Based upon the behavioral control method, authors in [47] successfully applied a strategy for multi-robot systems maintaining the global network integrity by putting constraints on the movement at each step and removing redundant connectivity. With the advantage of low computational complexity [42], [45], the potential field offers practical implementation in real robot systems across various environmental conditions, from free space [42] to environments with convex or concave obstacles [9], [46]. Inspired by biological swarms, the coordinated and synchronized motion of the robot swarm can effectively address unforeseen situations and complex environments with dense obstacles, such as forest-like settings [27], [48], [49]. However, the cave-like scenarios have not received much attention, i.e. the environment is not enough to maintain the desired configuration. Passing through a narrow environment, e.g. caves, corridors, narrow passages, and so on [50], [51] is still considered a crucial challenge for swarm navigation in general and formation control in particular, because maintaining a rigid formation [7] when moving through narrow spaces poses many risks of collision. As a result, local minima often occur when a robot formation moving through a narrow environment cannot maintain formation while avoiding collisions with the surrounding environment [52].

Within the domain of formation control, numerous approaches have been devised to facilitate the safe navigation of multi-robot formations through obstacle environments [12],

[31], [52]–[57]. This research domain is predominantly categorized into two primary types: *formation maintenance*, where the original shape remains unchanged or contract/expand the configuration over time [52]–[55]; and *formation transformation*, where the formation can transform to another configuration upon encountering obstacles [31], [56], [57].

In the context of *formation maintenance*, the formation consistently maintains its desired shape or realigns via shrink/expand configuration during movement. In [52], a model predictive control is employed to provide optimal control signals for each robot information based on the sharing of map updates. As a result, the amount of communication between them became significant, facing communication delays as the number of members in the formation increased. A region-based hierarchical control employed in [55] addresses obstacle-avoidance challenges in the virtual structure when moving through confined spaces. Here, robots move cohesively within a circular region, which can contract to avoid obstacles. Nevertheless, approaches in this case become more challenging in maintaining a predetermined formation in confined environments, as described in Fig. 3.1. This challenge arises significantly when the contraction of the formation heightens the risk of collisions among individuals and the surrounding space.

The second case, *formation transformation* involves the flexible transformation of the configuration into a distinct configuration to effectively navigate threats. The work in [57] introduces a formation maintenance and restructuring strategy combining behavioral control with obstacle velocity techniques to adeptly navigate dynamic environments. This restructuring strategy decided based on an auction-based market approach, presents an optimal solution for effective formation movement. However, communication costs are considerable, and communication between robots must be carried out continuously to make the optimal target decisions for each individual. An algorithm based on particle swarm optimization for the reconfigurable formation of multiple UAVs for visual inspection is introduced in [31]. The flight path is constructed, accounting for visual inspection constraints, and a reconfigurable topology is introduced to enable obstacle collision avoidance during operation. However, this centralized approach needs resources to obtain the optimal configuration at each time, which leads to challenges for robustness and scalability.

To the best of the authors' knowledge, our study constructs a strategy that can dynamically change formations based on calculations sensed from the environment. The changing decision signal is event-triggering and at a lower computational cost. Of these few existing studies, some studies use centralized control to construct control strategies [31], [53], [55], [56] or the information communicated between individuals is very large [52], [57]. Our solution is fully distributed control and the amount of communication information between individuals is lightweight, only including position and velocity information. In addition, when transforming to another configuration, the order of each member is calculated reasonably and is not pre-arranged, ensuring reconfiguration takes place safely and immediately.

Motivated by the limitations of the mentioned issues, this paper proposes an event-based deformation control strategy for a TVF to pass through confined space environments, as given in Fig. 3.1. The approach utilizes potential field techniques to design the behaviors of formation. The event-based mode-switching strategy is then presented based on the individual sensing capacity in formation. The main contributions of our work are threefold:

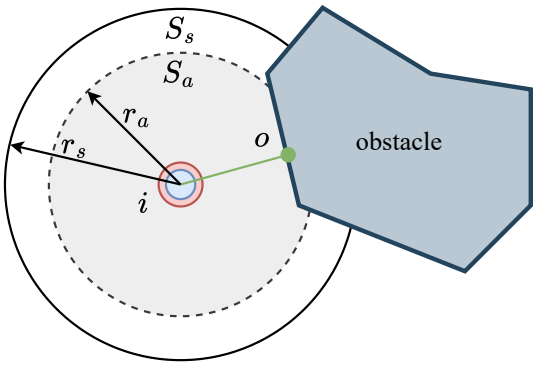


Figure 3.2: Illustration of a robot with a local range sensor. Each robot is equipped with a local sensor with sensing area S_s (solid while circle) being a circular disk within radius r_s. Additionally, alert area S_a (dashed gray circle) is a circular disk within radius r_a, with $r_a \leq r_s$, which is the zone that robot will active the repulsive force to avoid collision. The set $\mathcal{M}_i = \{o\}$ (green) is the nearest point from robot i to obstacle.

1. Propose an event-based deformation control strategy designed to enable the formation to scale, and transform to a straight line configuration, allowing it to effectively adapt to safely traverse confined environments;

2. Design behaviors for the TVF based on potential field function and demonstrate the stability of the control law through Lyapunov theorem;

3. Validate the proposed method through numerous simulations and comparisons. Software-in-the-loop (SIL) experiments are also conducted to evaluate the effectiveness.

The remaining sections of this paper are constructed as follows. Section 3.2 covers preliminaries and problem formulation. Section 3.3 presents the proposed distributed control strategy to reconfigure the formation to prevent collision with the environment. The proposed method is illustrated through simulation results and evaluation metrics in Section 3.4. Section 3.5 ends the paper with conclusions.

3.2 Preliminaries

3.2.1 Model of robots

Consider a swarm $\mathcal{N}$ that contains n robots labelled $i \in \{1, ..., n\}$. We model the swarm as a directed sensing graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where vertex set $\mathcal{V} = \{1, ..., n\}$ represents the robots, and edge set $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ contains the pairs of robots $(i, j) \in \mathcal{E}$ for which robot i can sense robot j. We denote $\mathcal{N}_i = \{j \in \mathcal{V} | \, (i, j) \in \mathcal{E}\} \subset \mathcal{V}$ as the set of n_i neighbours of a robot i in $\mathcal{G}$.

In this study, the robots' dynamics are expressed in discrete time. The position, velocity, and control input of robot i at time $t(k) = k\tau$ are denoted as $p_i(k), v_i(k), u_i(k) \in \mathbb{R}^3$, respectively, where τ represents the time step. The robots in the swarm are identical, each with a body radius r. Each robot i is equipped with an inertial measurement unit (IMU) to determine its position and orientation in a desired direction, a range sensor,

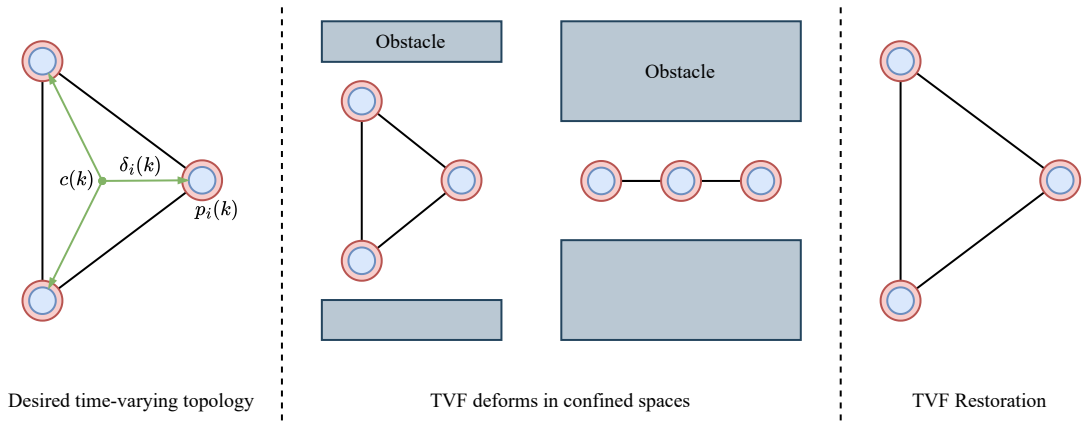


Figure 3.3: Schematic diagram of time-varying formation in the confined space

and a wireless ad-hoc network module for peer-to-peer communication with other robots. In this study, the communication delay between any two robots is assumed to be negligible [11], [58]. Additionally, each robot is equipped with a circular sensor [47] that offers a 360° field of view of the environment and can observed a maximum area S_s within a radius r_s, as depicted in Fig. 3.2. The set $\mathcal{M}_i(k) = \{o\}$ represents the observed obstacle points o which is the nearest point on the obstacle's boundary at time $t(k)$. Additionally, a circular with the radius r_a is denoted as an alert area that actives repulsive forces when robot i sensing obstacles or its neighbors.

According to [40], assuming that every robot in the swarm obeys a discrete linear system, given by

$$x_i(k+1) = A_i x_i(k) + B_i u_i(k) \tag{3.1}$$

where A_i and B_i are constant matrices. We consider the system to represent a robot with an underlying acceleration controller. The input u_i is an acceleration command, and the state $x_i = [p_i, v_i]^T \in \mathbb{R}^6$ is a vector containing the position and velocity. The velocities and accelerations of the robots are bounded by constant vectors, i.e. $\|v_i(k)\| \leq v_{\max}$ and $\|u_i(k)\| \leq u_{\max}$.

3.2.2 Problem formulation

This paper addresses the safe formation control of time-varying formation (TVF) in confined space. The primary challenges addressed are:

1. Ensuring collision-free navigation for the TVF, even when in the limited space environment.

2. Incorporating an event-triggering mechanism into the TVF control, allowing the formation to deform to another safe configuration.

Fig. 3.3 provides the general schematic diagram of the research problem tackled in this study. Initially, the desired configuration assigned to the TVF is denoted δ_0. At time k, the TVF can maintain the initial topology, i.e. $\delta(k) = \delta_0$. However, the confined space does not allow the formation maintaining topology δ_0 due to the potential collision. As a result, a new topology is triggered to be applied depending on the surrounding

space. Once the confined space is not detected by the robot, the initial topology δ_0 of the formation is restored. With the proposed control approach, the formation configuration can be effectively managed. During the movement of the TVF, collaborative and collision avoidance controls are driven by the APF term in the proposed control strategy.

Definition 1. *(Time-Varying Formation [40], [41] - TVF). Let $\delta(k) = [\delta_1(k), ..., \delta_n(k)]^T$ be a bounded time-varying vector that describes the desired formation configuration. The formation is said to achieve a TVF $\delta(k)$ if all robot i in the formation satisfy:*

$$\lim_{k\to\infty} \sum_{i=1}^{n} \|p_i(k) - \delta_i(k) - c(k)\| = 0, \quad \forall i \in \{1, ..., n\} \tag{3.2}$$

where $c(k)$ is called a formation center in the time k.

Definition 2. *(Safe Formation Control). Given the TVF as defined in Definition 1, the safe formation control of TVF is said to be realized for any robot i in the formation if the following conditions are simultaneously satisfied:*

1. *Formation configuration*

$$\lim_{k\to\infty} \sum_{i=0}^{n} \|(p_i(k) - \delta_i) - (p_j(k) - \delta_j)\| = 0 \tag{3.3}$$

   for all $i, j \in \{1, ..., n\}$, $i \neq j$.

2. *Safe distance between robots*

$$\|p_i(k) - p_j(k)\| > 2r \tag{3.4}$$

   for all $i \in \{1, ..., n\}$.

3. *Safe distance from obstacles*

$$\|p_i(k) - m\| > r \tag{3.5}$$

   for all $i \in \{1, ..., n\}$, and for all $m \in \mathcal{M}_i(k)$.

Remark 1. *According to the discrete linear model presented in (3.1), this study primarily focuses on the changes in the position and velocity status of the robot. Based on Definition 1, when (3.3) is established, robots reach the desired position, achieving the desired configuration. The establishment of (3.4) – (3.5) ensure that the robot will not collide with obstacles or other robots in the formation, respectively.*

3.2.3 Formation configurations

Formation configurations are emerging while robots are cooperating and interacting with the confined space environments. In this study, we categorize formation configurations into two primary configurations:

1. *Original configuration:* This formation configuration is maintained if any surrounding space is large enough to maintain the originally defined configuration or to transform it by contraction. The configuration can be a V-shape, a polygon, or others.

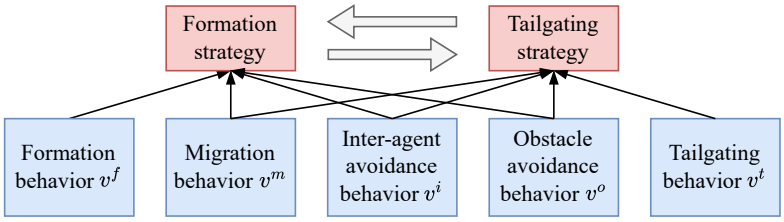


Figure 3.4: Overview of the proposed event-based deformation control. The proposed strategy is constructed by two primary emergent strategy, including *"Formation"* and *"Tailgating"*, which are highlighted by red boxes. There are five individual behaviors that contribute to the emergent strategy, which illustrated via blue boxes.

2. *Straight line configuration:* This formation configuration is performed when the robot does not have enough space to maintain the original configuration, forcing it convert to this form for safety assurance [57]. This configuration is constructed by having a robot follow and keep desired distance from its leader.

3.3 Event-based Deformation Control

To address the mentioned problem, the several potential force-based behaviors for each robot i are used to synthetic to the proposed controller, which contains emergent strategies to navigate TVF to deal with the narrow space environment while ensuring the shape and the safety of TVF, named *"Formation"*, and *"Tailgating"*. In *"Formation"* mode, the TVF maintains the original configuration, while in *"Tailgating"* mode, the TVF transforms to the straight line configuration, as presented in Section 3.2.3. The detail of the proposed strategies is described in Fig. 3.4.

3.3.1 Individual behaviors

According to [12], the potential field model is used to design individual behaviors a state-of-the-art model that allows robot swarm navigation in confined environments. From the original model, we design the rule of attraction for formation maintenance, tailgating and goal orientation; the rule of repulsion to prevent inter-robot collisions, and obstacle avoidance to avoid collisions with obstacles.

Firstly, to ensure goal-directed motion in environments, a target reaching behavior is provided by a preferred velocity vector. We denote v_{ref} is the preferred speed and u_{ref} is the preferred direction [44]. Then, the migration term, equal for every agent, corresponds to:

$$v_i^m = v_{\text{ref}} u_{\text{ref}} \tag{3.6}$$

Secondly, the formation behavior is designed as an attractive force that drive robots to move toward their desired positions. Denote $\kappa \in \mathbb{R}$ be the scale ratio, which can be determined in Section 3.3.2. Thus, a relative position-based controller of the form to maintain the desired shape, and enhance the abilities of TVF with scalable capabilities,

which is given as follow:

$$v_i^f = k_f \sum_{j=1}^{n} \left(p_j - p_i - \kappa\left(\delta_j - \delta_i\right)\right) \tag{3.7}$$

where $k_f > 0$ is the formation control gain.

Next, the tailgating behavior is designed based on the relative position between each robot with other robot in the TVF, and used to navigate TVF pass through the narrow environment. Let robot i follows robot l_i within desired distance d_{ref}, with $d_{\text{ref}} > 2r$, an attractive force field for robot i can be expressed as follows:

$$v_i^t = \begin{cases} k_t \left(p_{l_i} - p_i - d_{\text{ref}} \dfrac{v_{l_i}}{\|v_{l_i}\|} \right) + v_{l_i} & \text{if } l_i \neq -1 \\ 0 & \text{otherwise} \end{cases} \tag{3.8}$$

where k_t is the tailgating control gain. To determine the leader robot, the inner product $\tilde{p}_{ij}$ of the difference between robot j in the neighbor set $\mathcal{N}_i$ with robot i, $p_j - p_i$, and the desired direction, u_{ref}, is given as follows:

$$\tilde{p}_{ij} = \langle (p_j - p_i), u_{\text{ref}} \rangle \tag{3.9}$$

As a result, the value of $\tilde{p}_{ij}$ is positive, proving that robot j is in front of robot i according to u_{ref}, and vice versa. For all robots j in swarm, we obtain $\mathcal{P}_i = \{\tilde{p}_{ij}\}$. The leader robot l_i of robot i is chosen as the closest robot in front of it, i.e.

$$l_i = \begin{cases} \arg\min_j \{\tilde{p}_{ij} \in \mathcal{P}_i | \tilde{p}_{ij} \geq 0\} & \exists\ \tilde{p}_{ij} \geq 0 \\ -1 & \text{otherwise} \end{cases} \tag{3.10}$$

Finally, inter-agent avoidance and obstacle avoidance behaviors are also designed to prevent collision. Denote $\mathcal{M}_i$ be the set of finite points on the obstacle's boundary, which are the closest to robot i, as illustrated in Fig. 3.2. Those repulsive forces are given as follows:

$$v_i^i = k_i \sum_{j=1, j\neq i}^{n} v_{ij}^i \tag{3.11}$$

$$v_i^o = k_o \sum_{o \in \mathcal{M}} v_{io}^o \tag{3.12}$$

where $k_i, k_o > 0$ is the inter-agent collision and obstacle avoidance gains, respectively. Denote $p_{ij} = p_i - p_j$, and $\hat{p}_{ij} = \dfrac{p_{ij}}{\|p_{ij}\|}$ be the relation position and the normalized vector between robot i and robot j, respectively. Similarly, p_{io}, and $\hat{p}_{io}$ are the relative position and the normalized vector between robot i and obstacle o, respectively, with $o \in \mathcal{M}_i$. Thus, the associated inter-agent avoidance v_{ij}^i and obstacle avoidance v_{io}^o are given as follows:

$$v_{ij}^i = \begin{cases} \dfrac{r_a - \|p_{ij}\|}{r_a - 2r} \hat{p}_{ij} & \text{if } \|p_{ij}\| < r_a \\ 0 & \text{otherwise} \end{cases} \tag{3.13}$$

$$v_{io}^o = \begin{cases} \dfrac{r_a - \|p_{io}\|}{r_a - r} \hat{p}_{io} & \text{if } \|p_{io}\| < r_a \\ 0 & \text{otherwise} \end{cases} \tag{3.14}$$

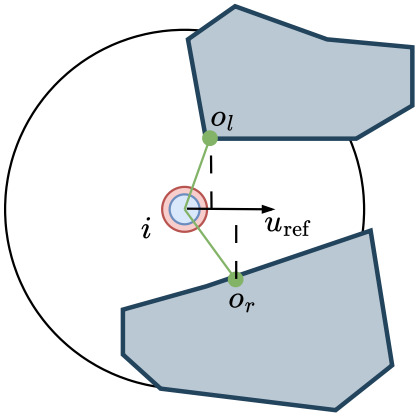


Figure 3.5: Environment's width estimation

3.3.2 Event-based Deformation Control strategy

As mentioned in Fig. 3.4, the proposed event-based deformation control includes two emergent strategies to navigate TVF safely through narrow space environment while maintaining the shape and ensuring the safety of swarm. At each time step, each robot determine the mode itself based on the sensing data collected from local sensor. The overall strategy can be summarised as follows:

$$\tilde{v}_i = \begin{cases} v_i^f + v_i^m + v_i^c + v_i^o & \text{if mode} = \textit{“Formation”} \\ v_i^t + v_i^m + v_i^c + v_i^o & \text{if mode} = \textit{“Tailgating”} \end{cases} \tag{3.15}$$

To search the large parameter space $\mathcal{K} = \{k_f, k_t, k_i, k_o\}$ of the controller, evolutionary optimization can be used for highest-order flight and lowest number of collisions. The evaluation of the swarm behavior is based on a single fitness function that sums three independent values, including order, agent-safety, and obs-safety, smaller or equal to 1 (ideal case). The fitness is determined in simulations where the swarm is initialized with random positions in an environment where obstacles are randomly placed. The description to seek parameter values are similar and detailed in the [12].

To select the suitable mode at each time step, an event-based mode switching is designed to deal with the requirement to navigate a TVF through a narrow space environment. According to the individual behaviors and emergent strategies mentioned above, the mode of each robot can be changed based on its sense with environment around.

Each obstacle point in the set $\mathcal{M}_i$ is clustered into two clusters on the left and right sides of the robot in u_{ref} direction. Denote o_l and o_r respectively be the nearest obstacle point on the left and right sides whose distance to the robot i is minimum. If o_l and o_r exist, the width of the environment can be estimated as follows:

$$w_e = \left| \left\langle \left(o_r - o_l \right), u_{\text{ref}} \right\rangle \right| \tag{3.16}$$

The estimation of the environment's width can be depicted in Fig. 3.5. Besides, the formation's width w_f can be easily determined through the predefined formation topology. Given the widths of the environment and formation, the scaling factor κ, which contributes to (3.7), can be computed as follows:

$$\kappa = \frac{w_e - 2r}{w_f} \tag{3.17}$$

As a result, each robot i can choose its mode and compute scaling factor κ. Denote $\lambda > 2$ be the threshold to transform between two mode. As presented in Algorithm 1, the desired velocity $\tilde{v}_i$ corresponding to its mode can be obtained.

Algorithm 1: Pseudocode of the EDC strategy

```
Get a set of observed obstacle points M_i;
if M_i is empty then
    mode ← "Formation";
    κ ← 1.0;
else
    Determine the obstacle points o_l, o_r;
    if ∄o_l or ∄o_r then
        mode ← "Formation";
        κ ← 1.0;
    else
        Get the space's width w_e;                          /* Eq. 3.16 */
        if w_e ≤ λr then
            mode ← "Tailgating";
        else
            mode ← "Formation";
            Estimate the desired formation width w_f;
            if w_e − 2r ≤ w_f then
                Compute the scaling factor κ;               /* Eq. 3.17 */
            else
                κ ← 1.0;
Compute desired velocity ṽ_i;                               /* Eq. 3.15 */
return ṽ_i;
```

After summing the contributions, we apply a cutoff on the acceleration at $u_{\max}$ according to

$$u_i = \frac{\tilde{u}_i}{\|\tilde{u}_i\|} \min(\|\tilde{u}_i\|, u_{\max}) \tag{3.18}$$

where $\tilde{u}_i(k+1) = (\tilde{v}_i(k+1) - \tilde{v}_i(k))/\tau$. Then, we apply a cutoff on the speed at $v_{\max}$, and get the velocity command v_i as follows:

$$v_i = \frac{\tilde{v}_i}{\|\tilde{v}_i\|} \min(\|\tilde{v}_i\|, v_{\max}) \tag{3.19}$$

3.3.3 Stability analysis

Theorem 1. *Given the TVF as described in (3.1), under the control law given in (3.15), the TVF asymptotically converges to the desired configuration.*

Proof. To prove Theorem 1, we consider the negligible effect of collision avoidance behaviors and also neglect the constant impact of migration behavior, for any time $t(k)$,

thus we conduct a stability analysis of the TVF in main contributions, i.e. formation and tailgating behaviors, which located in two separate modes, *"Formation"* and *"Tailgating"*, as mentioned in Section 3.2.3, which together ultimately lead to our conclusion.

Mode *"Formation"*: Denote L be the Laplacian matrix of the interaction topology graph of the swarm. Due to the fully connected, the entries of $L = [l_{ij}]$ are given as follows:

$$l_{ij} = \begin{cases} -1 & \text{if } i \neq j \\ n-1 & \text{if } i = j \end{cases} \tag{3.20}$$

The control law (3.7) can be shown as follows:

$$k_f \sum_{j=1}^{n} (p_j - p_i - \kappa\delta_{ji}) = k_f \sum_{j=1}^{n} (p_j - p_i) + b_i \tag{3.21}$$

where the bias $b_i = -k_f \sum_{j=1}^{n} \kappa\delta_{ji}$. In order to use the properties of the Laplacian matrix, the dynamics of the swarm system with the control law can be expressed as follows:

$$\dot{P} = -k_f LP + B \tag{3.22}$$

where $P = [p_1, ..., p_n]^T$, $B = [b_1, ..., b_n]^T$ and L is the Laplacian matrix of the interaction topology graph of the formation. For these values of its elements the Laplacian matrix has only one zero eigenvalue and the rest of its eigenvalues are positive and the same. Note also that the vector B is an eigenvector of L with the corresponding eigenvalue n and:

$$LB = nB \tag{3.23}$$

Defined the Lyapunov-like function for the TVF system as follows:

$$V_f = \frac{1}{2}\left(P - \frac{1}{k_f n}B\right)^T \left(P - \frac{1}{k_f n}B\right) \tag{3.24}$$

Talking the first derivative of V_f gives

$$\begin{aligned} \dot{V}_f &= \left(P - \frac{1}{k_f n}B\right)^T (-k_f LP + B) \\ &= -k_f \left(P - \frac{1}{k_f n}B\right)^T L \left(P - \frac{1}{k_f n}B\right) \leq 0 \end{aligned} \tag{3.25}$$

According to the LaSalle's invariance principle, it can be stated that as $k \to \infty$ the state P will converge to the largest invariant subset $\Omega = \left\{P | \dot{V}_f = 0\right\}$. In other words, the formation will close to the desired shape.

Mode *"Tailgating"*: In case robot i do not have it leader, we do not analysis the stability, because this robot do not contribute to the configuration maintenance of the TVF. Otherwise, in case robot i has its leader l_i, we define a candidate Lyapunov function as follows:

$$V_t = \frac{1}{2}\left(p_{l_i} - p_i - d_{\text{ref}}\frac{v_{l_i}}{\|v_{l_i}\|}\right)^T \left(p_{l_i} - p_i - d_{\text{ref}}\frac{v_{l_i}}{\|v_{l_i}\|}\right) \tag{3.26}$$

Taking the first derivative of V_t, gives:

$$\dot{V}_t = \left(p_{l_i} - p_i - d_{\text{ref}} \frac{v_{l_i}}{\|v_{l_i}\|} \right)^T (v_{l_i} - v_i) \tag{3.27}$$

By using the control law v_i^t in (3.8), $\dot{V}_t$ becomes:

$$\dot{V}_t = -k_t \left(p_{l_i} - p_i - d_{\text{ref}} \frac{v_{l_i}}{\|v_{l_i}\|} \right)^T \left(p_{l_i} - p_i - d_{\text{ref}} \frac{v_{l_i}}{\|v_{l_i}\|} \right) \leq 0 \tag{3.28}$$

Therefore, the Lyapunov stability is satisfied. That leads to robot i converges to its leader l_i and keep behind a distance d_{ref} along its leader direction. As a result, the straight line configuration can be guaranteed. □

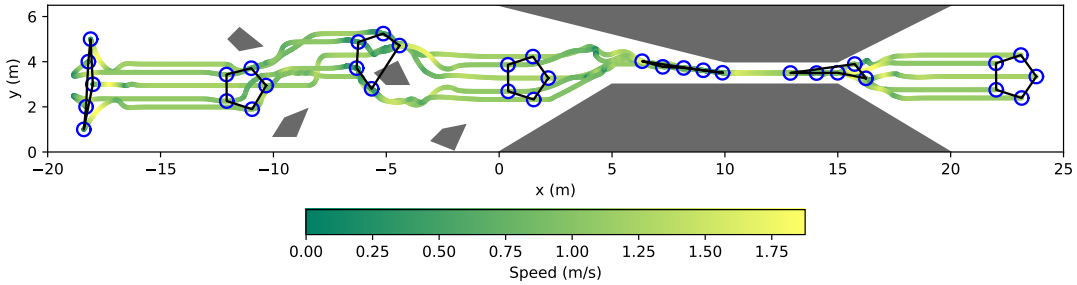
Remark 2. *In the research of this paper, decentralized control architecture is adopted for the TVF, i.e., each robot in the formation can decide the mode itself based on the information collected from surrounding environment. Correspondingly, the event-triggering condition we designed in Algorithm 1 is also distributed and presented in the form of compact sets. Besides, the control law (3.15) is demonstrated stability based on Lyapunov theory. Under the action of the designed control law, the TVF will asymptotically achieve the desired configuration in both modes.*

3.4 Results

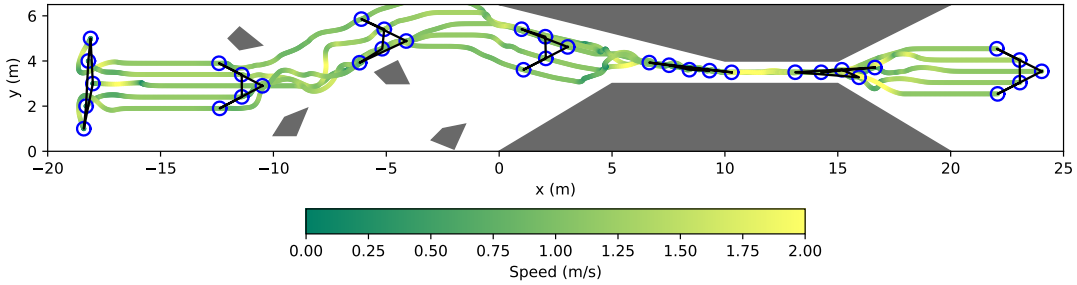
3.4.1 Simulation and Comparison

The proposed strategy is tested in a complex environment, which consists of 2 areas of different obstacle types, one area is the forest-liked environment, whose obstacle densities is 0.05 obs/m^2 (for -12 m $< x <$ 0 m), another area is the width-varying cave-like environment with the most narrowed space is 1 m (for 0 m $< x <$ 20 m). The TVF starts randomly from the left-hand side of the environment and has a mission to travel through the confined space to the right-hand side, with the desired direction $u_{\text{ref}} = [1, 0, 0]^T$. We set up a formation with 5 homogeneous robots with two formation configurations, including V-shape and polygon configurations, when the TVF transforms to *"Tailgating"* mode, the desired distance for a robot to follow its leader is set by $d_{\text{ref}} = 1$ m. They have constraints with $v_{\max} = 2$ m/s and $u_{\max} = 2$ m/s^2. The control period is set at $\tau = 0.1$ s. The comparison is done with pure behavior-based control (BC) [7], [12]. For comparison between the different methods, the following performance metrics are used: success rate, mean speed, and mean acceleration cost ($\sum \|u(k)\|^2/nT$), with T is the total travel time of the TVF. The parameters used in both strategies are the same to ensure fairness.

In the simulation, we examined how the *EDC* guided a TVF to navigate through confined spaces. The motion paths and corresponding velocity profile are presented in Fig. 3.6. Both two formations, including polygon and V-shape configurations, successfully navigate through confined spaces, which include forest-like and tunnel-like environments. Initially, all robots were on the left-hand side, which did not form the desired configuration. They move in the desired direction and form to the predefined shape. Whenever obstacles are detected, the robot activates the obstacle avoidance behavior, there is no narrow space

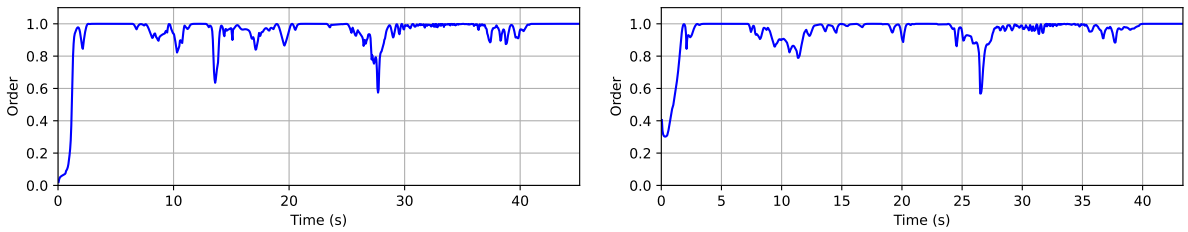


(a) Pentagon configuration



(b) V-shape configuration

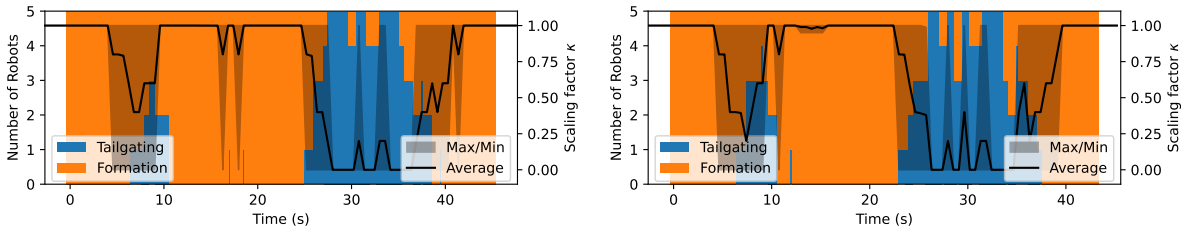
Figure 3.6: Motion paths and velocity profiles of the proposed *EDC* strategy in multiple configurations.



(a) Pentagon configuration

(b) V-shape configuration

Figure 3.7: The *order* values of the proposed *EDC* strategy



(a) Pentagon configuration

(b) V-shape configuration

Figure 3.8: Correlation of number of robots and scaling factor of the proposed *EDC* strategy

in the first area, and robots in the formation maintain the *"Formation"* mode. Once narrow space is observed, the TVF transforms to *"Tailgating"* mode, which is assigned itself by each robot observation. The straight line configuration is then created, which can help the formation pass through the gap. When the robot escapes the narrow passage, the mode turns back to *"Formation"* mode, which reforms to the desired configuration. Finally, the formation successfully passes through the confined space. Fig. 3.6 presents

Table 3.1: Comparison between *BC* and our method, *EDC*. Each comparison is over 10 simulations of 5 robots in two different configurations. The metrics displayed in the table are the success rate, mean speed, and mean acceleration cost.

Configuration	Strategy	Succ.	Mean speed (m/s)	Mean Acc. cost (m^2/s^4)
Pentagon	EDC	**10/10**	0.6773	**3.6873**
	BC	0/10	**0.7785**	4.3874
V-shape	EDC	**10/10**	0.7129	**4.0978**
	BC	0/10	**0.8877**	4.9402

the motion paths of the TVF, which are conducted by the proposed strategy in multiple configurations.

To further investigate the effectiveness of the proposed strategy, the *order* Φ metric is defined to measure the heading disturbance of robots in formation during movement. The order's values are in $[0, 1]$, and if the formation has no heading, the order is close to 1 [37].

$$\Phi = \frac{1}{n} \left\| \sum_{i=1}^{n} \frac{v_i}{\|v_i\|} \right\| \tag{3.29}$$

where v_i is the velocity of robot i.

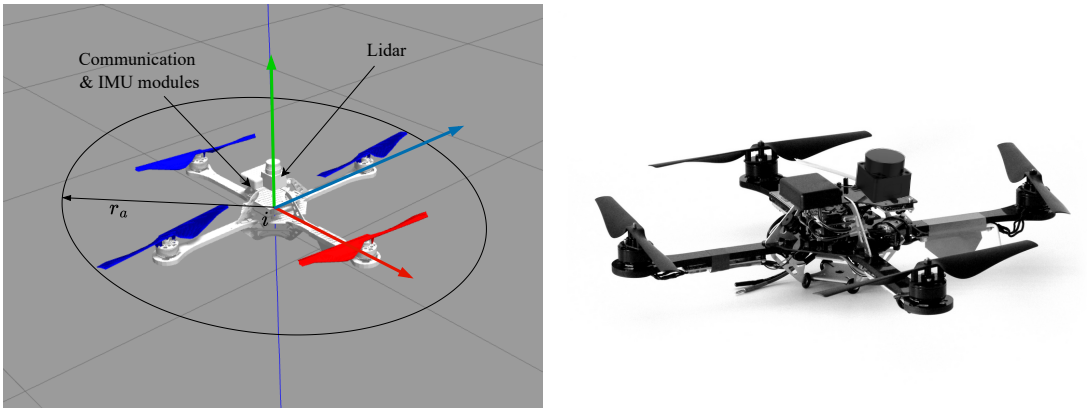
Fig. 3.7 illustrates the order information of the TVF's heading throughout the movement process. The figures reveal that the heading order of the swarm in both two scenarios during movement is satisfactory ($\Phi = 1$), When the formation encounters obstacles, the *order* value is changed but still forms the overall configuration. However, disorderliness in the heading order becomes apparent during the transition from *"Formation"* to *"Tailgating"* and vice versa. This is because the structure of the formation undergoes a significant change, resulting in a disorderly heading order.

Fig 3.8 describes the correlation between the number of robots in different modes and the scaling factor κ to evaluate the effectiveness of the synthesized controllers in the proposed deformation strategy. When the TVF encounters obstacles, the configuration is adapted based on the observation of each robot. As a result, the mode of each robot is different at the same time. Also, the scaling factor κ is different between robots in the TVF, due to its position and the observation. In collision-free, all robots in the TVF remain in *"Formation"* mode, which contributes to maintaining the original configuration. On the other hand, when narrow space is detected by all robots, the mode transforms to *"Tailgating"*, which forces robots to the straight line configuration to safely navigate through narrow space. The value of $\kappa = 0$ when all robots in the TVF are in the *"Tailgating"* mode.

To further verify the effectiveness of the proposed strategy against the pure behavioural-based control (BC) [7], [12], Table 3.1 presents a comparison between proposed strategy, *EDC* and BC. It can be seen that the proposed *EDC* strategy outperforms the *BC* in success rate, which can navigate formation passes through a confined space without any collision. Meanwhile, the *BC* always fails when encountering a narrow space (see Fig. 3.1). The mean acceleration cost per time is also smaller than the *BC*, because *BC* costs more energy to deal with the surrounding obstacles. The *EDC* provides an effective method to deal with obstacles by the adaptation configuration. However, the mean speed of the

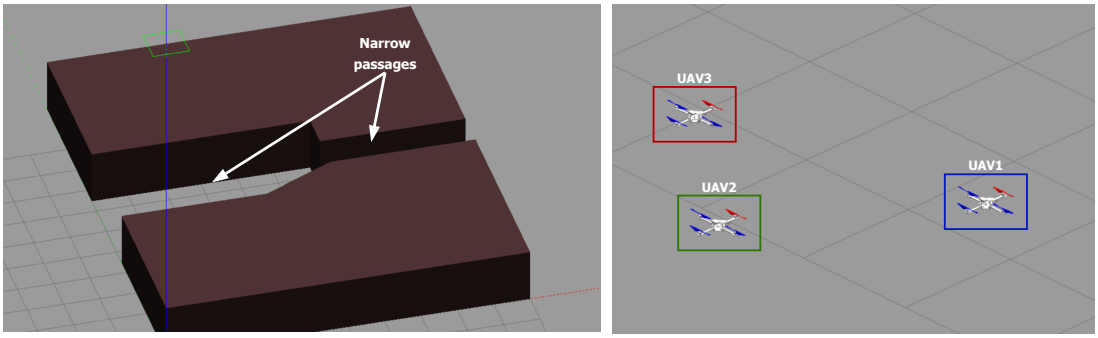
TVF is slightly smaller than the standard approach, due to the translation mode while moving affect to the speed down of the TVF.

3.4.2 Validation on the software-in-the-loop Gazebo



(a) Gazebo SIL model (b) Real UAV model [59], [60]

Figure 3.9: Used Hummingbird UAV model



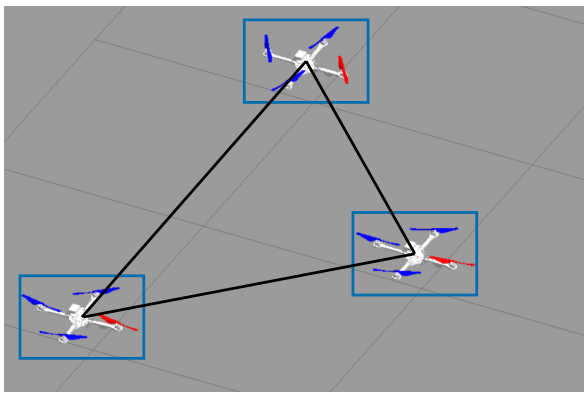
(a) The Gazebo environment (b) Random initial positions

Figure 3.10: The environment in SIL test

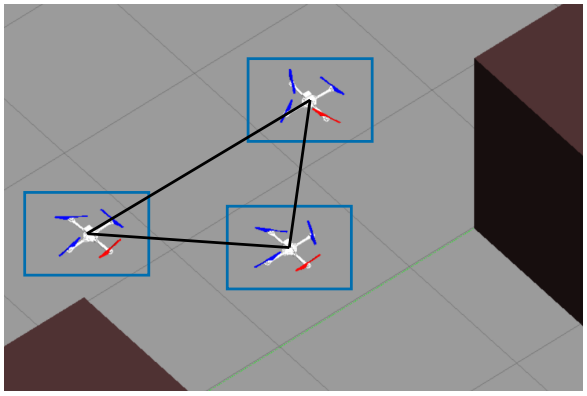
To evaluate the applicability of the proposed system, we have conducted a software-in-the-loop (SIL) validation that involves navigating UAV formation pass through a confined space, which only focuses on a narrow tunnel-like environment in a Gazebo[1]. The used UAV model is a Hummingbird quadrotor developed based on Gazebo-based RotorS simulator [59], as described in Fig. 3.9a. We set up three UAVs in the experiment with random positions, as shown in Fig. 3.10b. The environment in the SIL test includes two large obstacles, forming a width-varying tunnel, as depicted in Figure 3.10.

Fig. 3.11 presented the formation moving captured in the experiment. Moreover, the paths of three UAVs in the SIL test are recorded and are depicted in Fig. 3.12. At the beginning of the motion (step 1), three UAVs are in the random position. They then move to form the desired triangle formation (steps 375 - 750), as shown in Fig. 3.11a. When sensing the narrow passage, the formation shrinks to safely move through the passage (steps 1125 - 1500), as shown in Fig. 3.11b. When the narrow space is not suitable

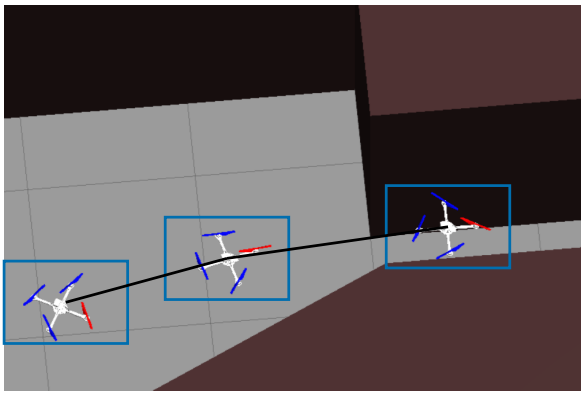
[1]Gazebo experiment video: https://youtu.be/AIAAzRiIepg



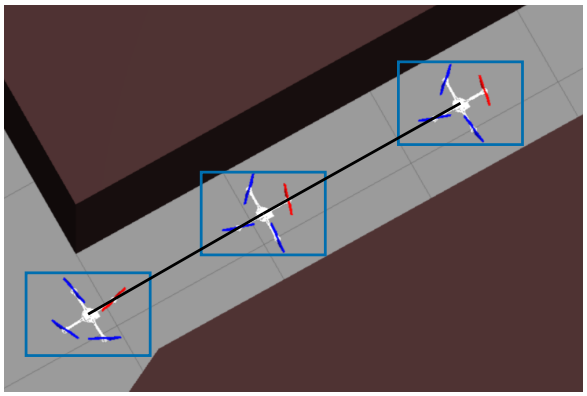
(a) Maintain triangle formation from random



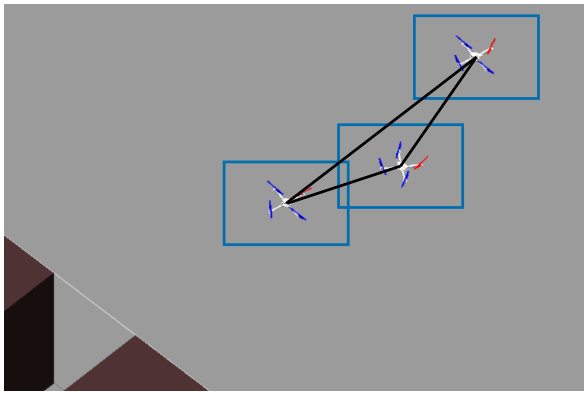
(b) Small-scaled triangle formation



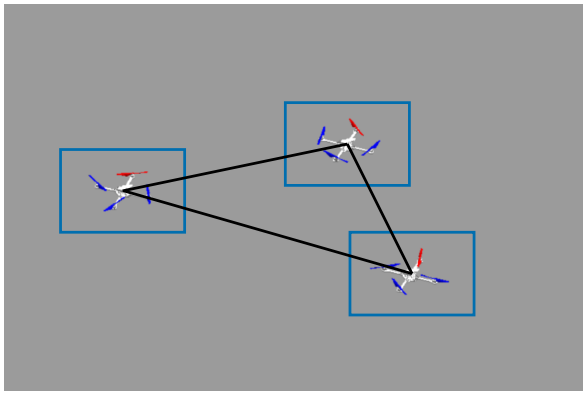
(c) Transform to line formation



(d) Line formation



(e) Transform back to triangle formation



(f) Original triangle formation

Figure 3.11: Validation results captured in the SIL Gazebo

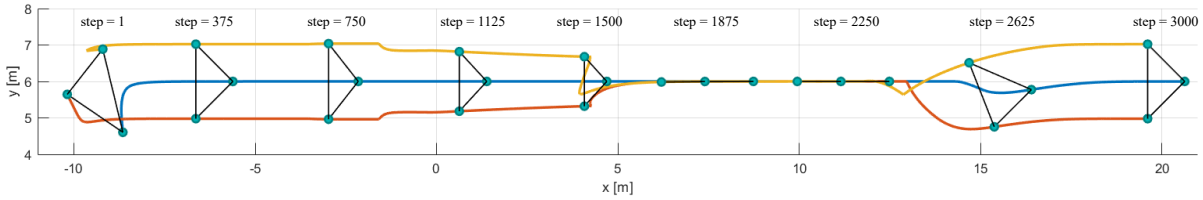


Figure 3.12: The recorded paths of three UAV in the SIL test (Top view)

for the original formation, the formation transforms to the straight line formation as Fig. 3.11c and then passes through the space with the line formation (steps 1875 - 2550), as Fig. 3.11d. Once the UAV senses enough space in the environment to maintain its original formation, the formation transforms back to the origin shape in Fig. 3.11e and moves to the target in Fig. 3.11f (steps 2625 - 3000). The experiment demonstrates that the proposed *EDC* strategy successfully navigates the UAVs' formation to the target by passing through the narrow passage.

3.5 Conclusion

In this paper, we have presented an event-based deformation control strategy to navigate a TVF through confined environments. The proposed strategy is constructed via several behaviors based on the potential field, which is synthesized together. The *EDC* strategy is proposed, which helps the TVF to safely pass through a confined space, especially in a narrow environment. The proposed strategy includes mode switching, which allows each robot to select the mode itself based on its observation. The proposed strategy is also proven stable via the Lyapunov theorem. Results and comparison with the standard method show the effectiveness of the proposed method in guiding the TVF travel safely through confined spaces. The conducted SIL test also confirms the performance of the *EDC* strategy.

Chapter 4

MPPDC: Model Prediction-based Perceptual Deformation Control for Multiple Robots in Narrow Space Environments

Published in:
Preprint, 2024

Abstract

Formation control plays a key role in multi-robot systems, in which a group of robots is coordinated to enhance task performance. Ensuring the safety of robot formations in clustered environments, especially in narrow space environments, remains a significant challenge. This paper proposes a Model Prediction-based Perceptual Deformation Control (MPPDC) strategy for a multi-robot swarm. The strategy proactively ensures the safety of robot formations moving through narrow space by changing the shape according to the environment observed from local sensors. The formation navigation method is modeled as objective functions that express the requirements, constraints, and limitations of the system in the formation movement mission. Additionally, a perceptual deformation strategy is also proposed for distributed decision-making. Finally, simulations, comparisons, and software-in-the-loop testing results are provided to demonstrate the effectiveness of the proposed approach.

Keywords: multi-robot system, distributed deformation control, model predictive control, narrow space environments

4.1 Introduction

Formation control of multiple robots enables various real-world applications such as mapping, construction, and search and rescue [1], [4]. These missions are often performed in harsh environments with clustered environments. To effectively cope with those tasks, it is crucial to ensure the safety of the robots both in their surrounding environments and with each other.

Natural formation collectives, such as school of fish or flocking of birds, reveal that coordinated navigation can be achieved through decentralized decision-making [61]. Their

motion can be explained by a set of simple rules based on the local information exchange, including repulsion that steers an agent away from its neighbors, cohesion that attracts the agent to the group, migration that orients its motion in a preferred direction, and additional repulsion from obstacles to avoid collisions with the environment [42]. Various swarm models based on the above rules have been implemented for swarm robots using artificial potential fields (APF) [7]–[9]. For instance, in [7], formation behaviors are integrated with other navigational behaviors to guide a robotic team to reach navigational goals, avoid collision, and maintain formation shape. Other research in [8] demonstrates the 3D collective behaviors with a swarm of fish-inspired miniature underwater robots using only implicit communication. However, achieving effective obstacle avoidance in limited space environments using potential field methods remains challenging for real robot swarms, as the density of obstacles can affect the robot's speed when navigating through such environments [62].

In practice, this issue can be addressed by tuning parameters (i.e., preferred speed, cohesion, repulsion, and other coefficients) specific to the environment and swarm configurations [10], [12]. Selecting the appropriate set of parameters is also challenging when the robot formation moves through different environments due to increased model complexity and the higher number of tunable parameters. In [10], a swarm robot model is proposed that controls the contraction/expansion of the robot formation using the APF method and an artificial neural network to optimize potential force parameters to adapt to environmental conditions. Experiments performed with three Turtlebot3 robots show that the proposed approach maintains formation distance and can adapt to the movement space, compared to the traditional APF approach. However, it is designed to create a formation structure that cannot navigate through narrow spaces. In [12], a communication-aware flocking control for a drone swarm is proposed using an evolutionary optimization framework to select appropriate order parameters and fitness functions to maximize the velocity and cohesion of the swarm. The drone moving in front must always promptly notify the drones behind to avoid crowding into the wall. Results demonstrated with thirty drones moving in tight formation and remaining within a limited area in which the drones avoided collisions with each other and with virtual obstacles by dividing and merging. The control algorithm shows the ability to arrange and expand formations in large spaces but does not demonstrate the ability of formation transitions in narrow environments. Parametric optimization APF-based approaches generally adjust the parameters of behavioral functions, forcing them to alter their formation shape according to environmental influences, but they cannot control the formation shape.

Alternative approaches using optimization for achieving navigation and/or formation control of a swarm have gained considerable attention in the last decade due to their significant impacts on swarm sustainability and their ability to address swarm constraints [63]. The works [13], [14] also demonstrated the remarkable potential of modern optimization-based motion planners for ensuring collision avoidance in multi-robot systems, although these planners are designed for individual point-to-point transitions and do not generate self-organized cohesive flight similar to biological swarms. Recent studies suggest that predictive controllers can improve the safety of robot swarms by predicting and optimizing the agents' future behavior in an iterative process [64], [65]. Model predictive control computes the control action of a system as the solution to an optimization problem that explicitly accounts for the robot dynamics and actuation constraints. The main limitation of MPC for performing these problems is its high computational cost [60], [66]. Fortu-

nately, the development of computational techniques and the support of various powerful libraries can handle this disadvantage in recent years [60], [64], [67].

In [64], a swarm navigation method using MPC for moving through a clustered environment is proposed and successfully implemented on a real drone system. Inspired by swarm behavior, the algorithm proposes corresponding cost functions and constraints for the robot swarm. The algorithm has proven its ability to navigate, maintain speed, and ensure safety. However, they are being deployed in a centralized manner, which is not feasible in practice moving far from the coordination center. An alternative distributed MPC version [68] is carried out with distributed computing implementation for homogeneous robots, demonstrating the possibility of implementing distributed MPC in a real system. Nonetheless, this approach still ensures that robots maintain a distance in large spaces without considering formation transitions in narrow environments. Maintaining formation while moving through tight environments poses a high risk of collision due to the conflict between preserving the original shape and avoiding obstacles.

These limitations result in the essential of an effective strategy to change the original shape of the formation to another to enhance the rational swarm's motion in narrow spaces. In other studies, strategies based on the structure of narrow environments involve distributed decision-making to enable a new shape of formation [9], [11]. In [11], the authors proposed formation change control to adapt to narrow environments with both static and dynamic obstacles using a set of target formation shapes. The approach optimizes parameters such as position, direction, and formation size using a consensus mechanism based on network-wide communication, i.e., a centralized solution based on the limited communication assumption. The formation shapes can be determined by a human designer or calculated automatically. The article employs a distributed consensus mechanism to compute the convex hull of robot positions and determine minimum/maximum positions in the desired movement direction for the entire swarm, subsequently assigning robots to positions within the formation. In this study, however, formation transition was achieved by reassigning robots to virtual points on the reference formation and designing collision-free trajectories to those points, without the robots self-configuring based on local interactions with the environment. Our previous work [46] also changed the shape of the V-shape formation based on the effect of the obstacle to maneuver the robot swarm to a narrow corridor. This V-shaped formation can be shrunk by closing the two wings of the V-shaped formation to the point of forming a straight line depending on the width of the narrow passage. Nevertheless, these approaches do not explicitly consider the physical limitations of the robots, which can result in infeasible control inputs and unexpected collisions in certain settings. Furthermore, the ability of each robot to make its own decisions based on environmental information is still limited.

Motivated by the consistent progress and technological advancements in applications of multi-robot formation in various real-world scenarios, in this paper, we aim to design an optimal deformation control strategy for a decentralized multi-robot team to ensure safety in narrow space environments. The contributions of the paper can be summarized as threefold:

1. A perceptual deformation control strategy is proposed to effectively navigate a multi-robot formation moving through narrow environments. Each robot is equipped with local sensors and communication modules to collect information from the surrounding environment and its neighbors for distributed decision-making. The formation

thus can be shrunk/expanded or transformed to the line formation according to the environment.

2. A model prediction-based control strategy is proposed to achieve the navigation requirements of maintaining formation, velocity, and direction, while effectively avoiding collisions. The proposed method models these objectives and constraints into fitness functions that can be easily expanded and deployed to multiple robots simultaneously.

3. The feasibility and effectiveness of the proposed strategy are demonstrated by simulation and comparison results. A software-in-the-loop experiment was also implemented using flying robots to validate abilities in real-world applications. We also release the source code of the proposed strategy.

The remaining sections of this paper are organized as follows. Section 4.2 describes the formation model. Section 4.3 presents the proposed model prediction-based perceptual deformation control strategy to navigate the multi-robot formation for ensuring collision avoidance in the narrow space environment. Simulation results, comparisons, and a software-in-the-loop experimental validation using aerial robots are given in Section 4.4 to highlight the feasibility and efficiency of the proposed strategy. The paper ends with conclusions drawn in Section 4.5.

4.2 Background

Consider a swarm $\mathcal{N}$ that contains N robots labelled $i \in \{1, ..., N\}$. We model the swarm as a directed sensing graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where vertex set $\mathcal{V} = \{1, ..., N\}$ represents the robots, and edge set $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$ contains the pairs of robots $(i, j) \in \mathcal{E}$ for which robot i can sense robot j. We denote $\mathcal{N}_i = \{j \in \mathcal{V} | \, (i, j) \in \mathcal{E}\} \subset \mathcal{V}$ as the set of N_i neighbours of a robot i in $\mathcal{G}$.

In our study, the dynamics of the robots are represented in discrete time. Denote $p_i(k), v_i(k), u_i(k) \in \mathbb{R}^3$ respectively be the position, velocity and control input of the robot i at the time $t(k) = k\tau$, where τ is the time step. The robots in the swarm are homogeneous with a body radius r. Each robot i is equipped with an inertial measurement unit (IMU) to determine its position and orientation to a desired direction, a range sensor, and a wireless ad-hoc network module to carry out peer-to-peer communication with other robots. In this work, the communication delay between each pair of robots is negligible [11], [58]. The local sensor is equipped on each robot to provide a 360° field of view of the surrounding environment and can scan a maximum area S_s within radius r_s, as shown in Fig. 4.1. A set $\mathcal{M}_i(k) = \{m\}$ represents the observed obstacle points at time $t(k)$.

According to [64], assuming that every robot in the swarm obeys a discrete linear system, given by

$$x_i(k+1) = A_i x_i(k) + B_i u_i(k) \tag{4.1}$$

where A_i and B_i are constant matrices. We consider the system to represent a robot with an underlying acceleration controller. The input u_i is an acceleration command, and the state $x_i = [p_i; v_i] \in \mathbb{R}^6$ is a vector containing the position and velocity. The velocities and

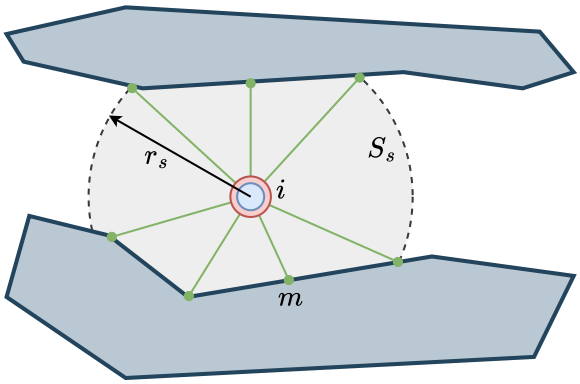


Figure 4.1: Illustration of a robot with a local range sensor. Each robot is equipped with a local sensor with sensing area S_s (dashed gray circle) being a circular disk with radius r_s. The set $\mathcal{M}_i = \{m\}$ (green) is the observed point cloud data.

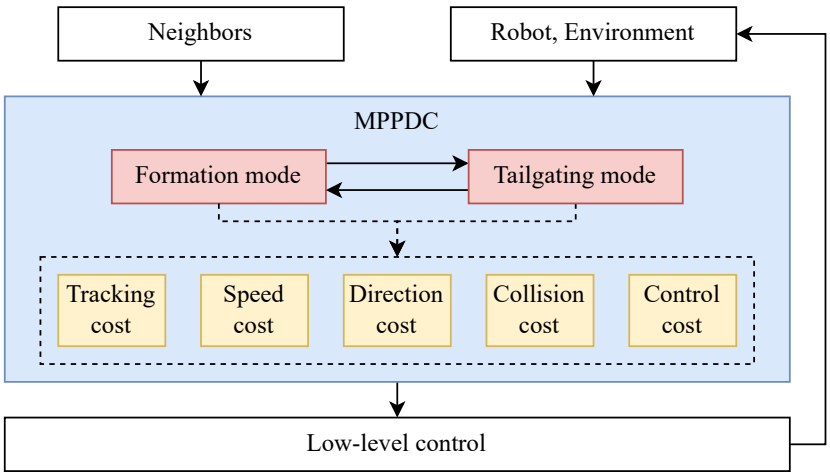


Figure 4.2: The diagram of the proposed method implemented for each robot. The point cloud data of the environment observed from local sensors and the information from the neighbors are collected as inputs to the proposed approach. Our predictive control strategy, which includes two primary modes: *"Formation"* and *"Tailgating"*, is formulated by the weighted-sum of five cost functions to address the requirements *(O1-O4)*. The optimal solver is constructed to generate the control signal for the low-level controller.

accelerations of the robots are bounded by constant vectors, i.e. $v_{\min} \leq v_i(k) \leq v_{\max}$ and $u_{\min} \leq u_i(k) \leq u_{\max}$.

4.3 MPPDC: Model Prediction-based Perceptual Deformation Control Strategy

In this work, the formation of robots has the mission to pass through a narrow environment. During the movement, the formation is required to *(O1)* maintain their shape; *(O2)* move along to the predefined direction $u_{\text{ref}} \in \mathbb{R}^3$, where $\|u_{\text{ref}}\| = 1$, which can be determined by information collected from surrounding environments [9], [69], with *(O3)* the desired speed $\bar{v}_{\text{ref}} \in \mathbb{R}$; and *(O4)* ensuring no collision with other neighbors or obstacles in the environment.

The proposed deformation control strategy is illustrated in Fig. 4.2, which contains two modes, *"Formation"* and *"Tailgating"*. Based on the point cloud data obtained from a local sensor equipped on each robot, the formation can be switched from *"Formation"* to *"Tailgating"* mode if the width of the environment is too narrow to maintain the whole formation topology. In *"Formation"* mode, the formation can maintain its original shape, and in *"Tailgating"* mode, the formation transforms into straight line shapes with positions determined based on their orientation relative to the specified direction u_{ref}. The straight line shape is chosen in our work because it is an effective shape that can be used to go through obstacles like narrow valleys or tunnels [57].

In the *"Formation"* mode, the aim is to maintain the desired given shape. Let $\delta_{ij} \in \mathbb{R}^3, \forall j \in \mathcal{N}_i$ be the vector representing the desired position of robot i with respect to neighbor j. Note that $\delta_{ji} = -\delta_{ij}$ must be satisfied for all pairs (i, j), $j \neq i$, for consistency and feasibility of the formation. Note also that $\delta_{ii} = 0$ for all i. This aim can be given as follows [41], [70]:

$$\lim_{k\to\infty} \left(p_j(k) - p_i(k) + \kappa\delta_{ij}\right) = 0, \quad \forall i, j \in \{1, ..., N\}, i \neq j \tag{4.2}$$

where $\kappa \in [0, 1]$ is the scaling factor representing the shrinkage of the formation. This scaling factor can be determined by the environment's width in Section 4.3.2. As a result, the desired relative position of every robot i in the formation can be described as follows:

$$p_i^*(k) = \frac{1}{N_i} \sum_{j\in\mathcal{N}_i} \left(p_j\left(k\right) + \kappa\delta_{ij}\right) \tag{4.3}$$

In addition, to determine the leader robot, the inner product $\tilde{p}_{ij}$ of the difference between robot j in the neighbor set $\mathcal{N}_i$ with robot i, $p_j - p_i$, and the desired direction, u_{ref}, is given as follows:

$$\tilde{p}_{ij} = \langle (p_j - p_i), u_{\text{ref}} \rangle \tag{4.4}$$

As a result, the value of $\tilde{p}_{ij}$ is positive, proving that robot j is in front of robot i according to u_{ref}, and vice versa. For all robots j in the neighbor set $\mathcal{N}_i$, we obtain $\mathcal{P}_i = \{\tilde{p}_{ij}\}$. The leader robot l_i of robot i is chosen as the closest robot in front of it, i.e.

$$l_i = \begin{cases} \arg\min_j \{\tilde{p}_{ij} \in \mathcal{P}_i | \tilde{p}_{ij} \geq 0\} & \exists\ \tilde{p}_{ij} \geq 0 \\ -1 & \text{otherwise} \end{cases} \tag{4.5}$$

The leader selection process is presented in Alg. 2.

Algorithm 2: Pseudocode of leader selection

```
foreach j ∈ N_i do
    Compute inner product p̃_ij;        /* Eq. 4.4 */
    P_i ← p̃_ij;
Select the leader l_i to follow;        /* Eq. 4.5 */
return l_i;
```

On the other hand, the *"Tailgating"* mode allows robot i to follow robot j and keeps a distance $\bar{d}_{\text{ref}} \in \mathbb{R}$ with robot j, to form the straight line shape. This mode is enabled

when the operation space becomes too narrow to maintain the original formation as the inter-agent collision might occur. At that time, the tailgating strategy is essential for the swarm so that each robot can move one after another through the narrow passage without collision. Assume robot i follows robot j with the desired distance $\bar{d}_{\text{ref}}$. The following conditions must be satisfied:

$$\lim_{k\to\infty} \|p_j(k) - p_i(k)\| = \bar{d}_{\text{ref}} \tag{4.6}$$

In order to allow robot i to follow the robot j and keep a desired distance d_{ref} with robot j, the desired relative position of robot i can be expressed as follows:

$$p_i^*(k) = p_j(k) - \bar{d}_{\text{ref}} u_{\text{ref}} \tag{4.7}$$

Therefore, the requirement of shape maintenance *(O1)*, i.e. formation maintenance in *"Formation"* mode and following in *"Tailgating"* mode, can be synchronized into a tracking objective that is represented by desired position p_i^* to reduce the complexity of the optimization process.

4.3.1 The predictive control formulation

This section describes our proposed predictive strategy. Denote $P \in \mathbb{N}^+$ as the prediction horizon, which is finite and shifts forward at every time step. Let $(\cdot)(k+l|k)$ be the predicted value of $(\cdot)(k+l)$ with the information available at time $t(k)$ and $l \in \{0, ..., P\}$. Then, the condition of the robot state is written as follows:

$$x_i(k|k) = x_i(k) \tag{4.8}$$

Let $X_i(k) \in \mathbb{R}^{6P}$ be the stacked sequence of the predicted states $x_i(k+l|k)$ over the horizon $l \in \{1, ..., P\}$ and $U_i(k) \in \mathbb{R}^{3P}$ the stacked sequence of the predicted control inputs $u_i(k)$ over the horizon $l \in \{0, ..., P-1\}$. To resolve the formation requirements, we modeled those requirements into candidate cost functions, i.e. *(O1)* tracking cost $J_{t,i}(k)$, *(O2)* speed cost $J_{s,i}(k)$, *(O3)* direction cost $J_{d,i}(k)$, *(O4)* collision costs, including obstacle avoidance cost $J_{o,i}(k)$, and inter-agent collision cost $J_{i,i}(k)$, and an addition control cost $J_{u,i}(k)$ is a penalty term to get the minimal solution. Hence, the distributed formation control system can be modeled by the non-convex optimization problem as follows:

$$\begin{aligned} \min_{U_i(k)} \big(&J_{t,i}(k) + J_{s,i}(k) + J_{d,i}(k) + \\ &J_{o,i}(k) + J_{i,i}(k) + J_{u,i}(k)\big) \end{aligned} \tag{4.9}$$

subject to:

$$\begin{aligned} &x_i(k+l+1|k) = Ax_i(k+l|k) + Bu_i(k+l|k), \\ &x_i(k|k) = x(k), \\ &v_{\min} \le v_i(k+l|k) \le v_{\max}, \\ &u_{\min} \le u_i(k+l|k) \le u_{\max}, \end{aligned} \tag{4.10}$$

with $l \in \{1, ..., P\}$, and $i \in \mathcal{N}$.

Tracking term

The tracking term for robot i at time $t(k)$ aims to optimize the following square error between the desired position p_i^* and the predicted position p_i, thus, the robot i is encouraged to closely track its desired position, leading to more accurate and reliable path following. This term is defined as follows:

$$J_{t,i}(k) = w_t \sum_{l=1}^{P} \|p_i^*(k+l|k) - p_i(k+l|k)\|^2 \tag{4.11}$$

where w_t is the positive tracking weight.

Speed term

The speed term is used to maintain smooth, consistent motion at the desired speed $\bar{v}_{\text{ref}}$, which is crucial for efficiency and safety in environments. Any deviation from the desired speed is penalized by squaring the difference between the actual and desired speed values, which amplifies larger errors. The term allows the robot to adjust its speeds to stay as close as possible to $\bar{v}_{\text{ref}}$. It is given as follows:

$$J_{s,i}(k) = w_s \sum_{l=1}^{P} \left(\|v_i(k+l|k)\|^2 - \bar{v}_{\text{ref}}^2 \right)^2 \tag{4.12}$$

where w_s is the positive propulsion weight.

Direction term

The direction term used to navigate the robot along the desired direction u_{ref}. To evaluate how close the robot is to the target direction, the dot product between the predicted velocity and the desired direction u_{ref} is calculated and normalized by the magnitude of the velocity. If the velocity perfectly aligns with the reference direction, this term will be close to zero; otherwise, the value increases, indicating misalignment. By minimizing this term, the robot is encouraged to follow the desired path more closely, ensuring smooth and effective directional control. It is given as follows:

$$J_{d,i}(k) = w_d \sum_{l=1}^{P} \left(1 - \frac{\langle v_i\left(k+l|k\right), u_{\text{ref}} \rangle^2}{\|v_i(k+l|k)\|^2} \right)^2 \tag{4.13}$$

where w_d is the positive direction weight.

Collision avoidance term

Let $d_{ij} = \|p_j - p_i\|$ be the distance between the center of two robots i and j, and d_{im} be the distance of robot i to the obstacle point m. To prevent the robots from colliding with their neighbors or the obstacles, the collision avoidance constraints [64], [71], [72] are defined as follows:

$$\begin{aligned} d_{ij}(k+l|k) &\geq 2r \quad i \in \mathcal{N}, j \in \mathcal{N}_i \\ d_{im}(k+l|k) &\geq r \quad i \in \mathcal{N}, m \in \mathcal{M}_i(k) \end{aligned} \tag{4.14}$$

where the safety distance between two robot's positions must be larger than $2r$ and the distance to the obstacle set $\mathcal{M}_i(k)$ must be greater than robot radius r at time $t(k)$ that the robot should keep. In an unknown environment, the number of obstacle points in obstacle set $\mathcal{M}_i$ and the number of neighbors at time $t(k)$ are different. It is, therefore, difficult to model the safety term via hard constraints. In this study, alternative cost functions are used to overcome this problem.

Inspired by [73], [74], the obstacle avoidance cost function is a logistic function given as follows:

$$J_{o,i}(k) = w_o \sum_{l=1}^{P} \frac{1}{1 + \exp\left(\alpha \left(d_{im}^{\min}(k+l|k) - r\right)\right)} \tag{4.15}$$

where $w_o > 0$ is a constant weight, $\alpha > 0$ is a parameter representing the smoothness of the cost function, and

$$d_{im}^{\min}(k+l|k) = \min\left\{d_{im}(k+l|k) | m \in \mathcal{M}_i\right\} \tag{4.16}$$

Moreover, the inter-agent collision cost function is also defined as follows:

$$J_{i,i}(k) = \frac{w_i}{N_i} \sum_{l=1}^{P} \sum_{j \in \mathcal{N}_i} F_{ij}(k+l|k) \tag{4.17}$$

with

$$F_{ij}(k+l|k) = \begin{cases} 0 & \text{if } d_{ij}(k+l|k) \geq \beta r \\ \dfrac{\beta r - d_{ij}(k+l|k)}{(\beta - 2)r} & \text{if } 2r < d_{ij}(k+l|k) < \beta r \\ \infty & \text{if } d_{ij}(k+l|k) \leq 2r \end{cases} \tag{4.18}$$

where $w_i > 0$ is the constant inter-agent weight, $\beta > 2$ is the influence ratio of the neighbors [7].

Control effort

The control cost is used as a penalty part to get the minimal control signal, which is given as follows:

$$J_{u,i}(k) = w_u \sum_{l=0}^{P-1} \left\|u_i(k+l|k)\right\|^2 \tag{4.19}$$

where $w_u > 0$ represents the constant control weight.

4.3.2 Perceptual deformation control strategy

In this section, details about making the state-changing decision and computing the scaling factor κ of the desired formation are presented. To estimate the environment's width, each robot i only considers the obstacle point set, $\mathcal{M}_{ui}$, in the u_{ref} direction. It is given by

$$\mathcal{M}_{ui} = \left\{\mathcal{M}_i | \left\langle \left(p_i - \mathcal{M}_i\right), u_{\text{ref}} \right\rangle < 0\right\} \tag{4.20}$$

By using the DBSCAN [75] technique, the obstacle point set $\mathcal{M}_{ui}$ is clustered into two clusters on the left and right sides of the robot in u_{ref} direction. Denote $\mathcal{M}_{i,l}$ and $\mathcal{M}_{i,r}$

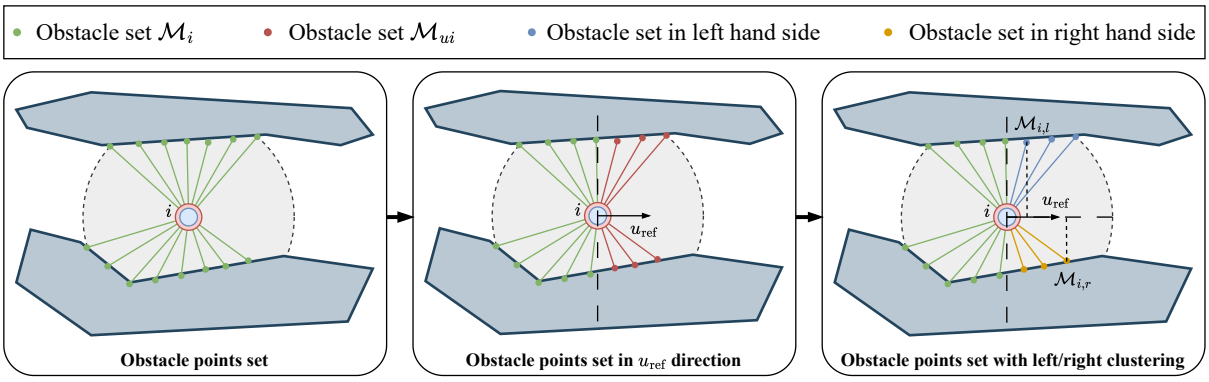


Figure 4.3: The process of estimating the environment's width: the robot first obtains point cloud data $\mathcal{M}_i$ from the local sensor (green). It then selects a point set $\mathcal{M}_{ui}$ (red) that is in front of the robot in a predefined direction u_{ref}. It then divides those points into two sub-sets on the left-hand (blue) and right-hand (yellow) sides. Finally, the points that are nearest to the robot in u_{ref} direction in each side, $\mathcal{M}_{i,l}$, and $\mathcal{M}_{i,r}$ are selected to estimate the width of the environment.

respectively as the obstacle point on the left and right sides whose distance to the robot i is minimum. The width of the environment can be estimated as follows:

$$w_e = |\langle (\mathcal{M}_{i,r} - \mathcal{M}_{i,l}), u_{\text{ref}} \rangle| \tag{4.21}$$

The estimation of the environment's width can be depicted in Fig. 4.3 and described in Alg. 3. Besides, the formation's width w_f can be easily determined through the predefined formation topology. Given the widths of the environment and formation, the scaling factor κ can be computed as follows:

$$\kappa = \frac{w_e - 2r}{w_f} \tag{4.22}$$

Algorithm 3: Pseudocode to estimate the environment's width

1 Get obstacle point set $\mathcal{M}_{ui}$ in front of robot i in motion direction u_{ref}; `/* Eq. 4.20 */`
2 Cluster obstacle point set $\mathcal{M}_{ui}$ to left and right hands of robot using DBSCAN;
3 Find the pair of obstacle points $(\mathcal{M}_{i,l}, \mathcal{M}_{i,r})$, whose distance to u_{ref} is minimum;
4 Obtain the environment's width w_e; `/* Eq. 4.21 */`
5 **return** w_e;

As a result, each robot i can choose its mode and compute scaling factor κ. It then computes the desired trajectory corresponding to its mode and the optimal control signal using MPC, as presented in Alg. 4. This problem can be solved be commonly used, nonlinear programming (NLP) solvers, such as the Sequential Least Squares Programming (SLSQP) [76]. We implemented the MPC model in Python with the help of a state-of-the-art optimization software library [67] to find the solution to the problem.

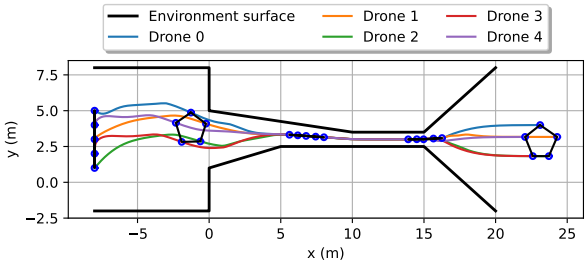
Algorithm 4: Pseudocode of the MPPDC strategy

```
Get the observed obstacle points M_i;
if M_i is empty then
    mode ← "Formation";
    κ ← 1.0;
else
    Get the environment's width w_e;                          /* Alg. 3 */
    if w_e is None then
        mode ← "Formation";
        κ ← 1.0;
    else
        if w_e ≤ λr then
            mode ← "Tailgating";
        else
            mode ← "Formation";
            Estimate the desired formation width w_f;
            if w_e − 2r ≤ w_f then
                Compute the scaling factor κ;                 /* Eq. 4.22 */
            else
                κ ← 1.0;
switch mode do
    case "Formation" do
        Get the desired trajectory p_i^*;                     /* Eq. 4.3 */
    case "Tailgating" do
        Select leader to follow;                              /* Alg. 2 */
        Get the desired trajectory p_i^*;                     /* Eq. 4.7 */
Formulate the cost function and solve the MPC problem to obtain the optimal
    control signal u_i^*;                                     /* Eqs. 4.9-4.10 */
return u_i^*;
```

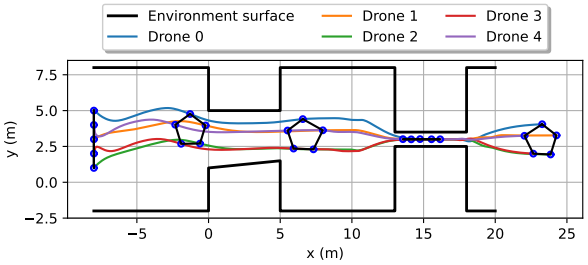
4.4 Results and Discussion

In this section, our proposed model prediction-based perceptual deformation control strategy *(MPPDC)*, is examined and evaluated through simulation experiments in various cave-like scenarios as depicted in Fig. 4.4 and compares with the behavior-based deformation control strategy *(BDC)*, which is reconstructed based on the original model [7], [12], and the model prediction-based formation control method *(MPFC)* [64], [68]. For comparison between the different methods, the following performance metrics are used: success rate, mean *order* Φ, mean speed (m/s), mean formation error ε (m), and acceleration cost $(\sum \|u(k)\|^2/N)$ $(\mathrm{m}^2/\mathrm{s}^3)$ [77].

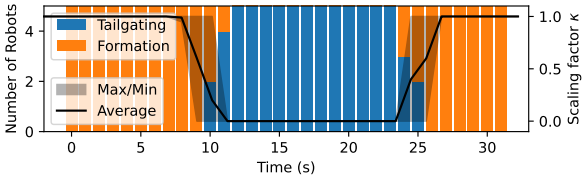
The robot swarm has five homogeneous individuals with size $r = 0.2$ m, equipped with a local sensor with a maximum sensing range of $r_s = 3$ m. The maximum speed



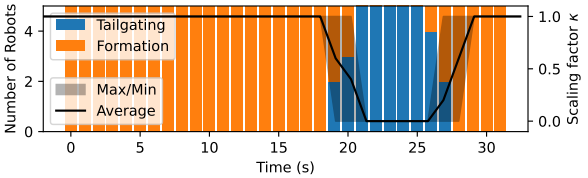
(a) Scenario 1 - Motion paths



(b) Scenario 2 - Motion paths



(c) Scenario 1 - Correlation of number of robots and scaling factor



(d) Scenario 2 - Correlation of number of robots and scaling factor

Figure 4.4: The motion path of a pentagon formation in two scenarios using the proposed strategy.

is $\|v_{\max}\| = 1.5$ m/s and the maximum control is $\|u_{\max}\| = 2.0$ m/s^2. In both scenarios, the desired velocity during movement is set to $\bar{v}_{\text{ref}} = 1$ m/s. The desired formation is set as a regular pentagon. Initially, all robots are randomly positioned on one side of the zone. The swarm then navigates through the narrow passages in the zone along the desired direction $u_{\text{ref}} = [1, 0, 0]^T$. The mission is considered complete when all the robots have crossed to the opposite side of the region, specifically $p_x \geq 22.0$ in all scenarios. The control period of robots is set at $\tau = 0.1$ s.

The *order* Φ metric [37] is used to measure the heading consensus of robots in formation during movement. The order value is in $[0, 1]$. If robots in the formation have the same direction, the order value is close to 1. It is defined as follows.

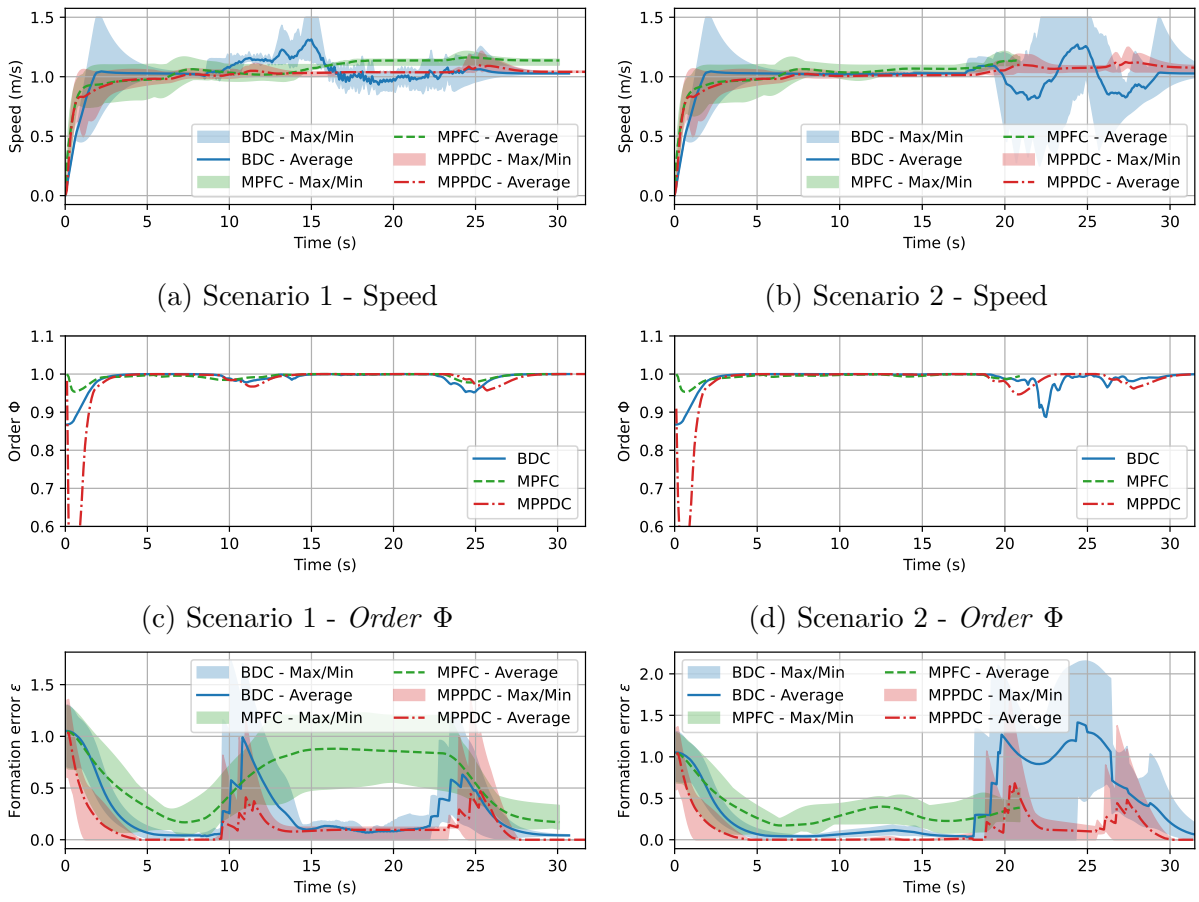
$$\Phi = \frac{1}{N} \left\| \sum_{i=1}^{N} \frac{v_i}{\|v_i\|} \right\| \tag{4.23}$$

To demonstrate the effectiveness in maintaining the formation, the *formation error* evaluates the deviation between the actual position p_i and its expected position p_i^* of all robots in the formation to demonstrate the effectiveness of the formation control strategy [70]. The closer this error is to zero, the more effective the controller becomes. The formation error ε_i of robot i can be calculated as follows:

$$\varepsilon_i = \|p_i - p_i^*\| \tag{4.24}$$

4.4.1 Results and Comparisons

According to Fig. 4.4, in both scenarios, the robot's formation successfully passes through the narrow space. At the beginning of the motion, the formation rapidly forms a desired pentagon shape. They then change their shape to safely pass through the tight passages, and then transform back to the desired shape after escaping the passages.



(a) Scenario 1 - Speed

(b) Scenario 2 - Speed

(c) Scenario 1 - *Order* Φ

(d) Scenario 2 - *Order* Φ

(e) Scenario 1 - *Formation error* ε

(f) Scenario 2 - *Formation error* ε

Figure 4.5: Comparison results of three strategies in two scenarios.

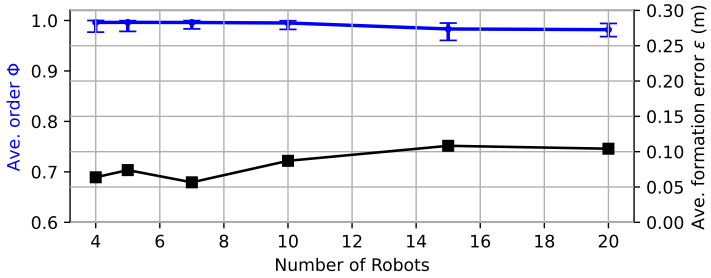


Figure 4.6: Effect of number of robots on system performance, including *order* Φ value and formation error ε.

Fig. 4.4c-4.4d depict the correlation of the number of robots in different modes and scaling factor κ. It can be seen that our strategy provides distributed decision-making capabilities based on the collected information. In particular, the modes of the robot swarm at a specific time vary depending on the structure of the environment and its neighbors.

Besides, when the environment is large enough to maintain the original shape, the value of scaling factor κ is determined to be 1. When the width of the environment is decreased, the κ of each robot is also determined to be smaller than 1 and gradually decreases depending on the width of the passage, adjusting the shape of the formation to shrink. Once the mode is switched to *"Tailgating"*, the κ value is determined to be 0. Similarly, when the formation exits the narrow corridor, the width of the environment

Table 4.1: The comparison between *BDC*, *MPFC*, and the proposed *MPPDC*. Each comparison is over 10 simulations of 5 robots in two different scenarios. The metrics displayed in the table are the success rate, mean *order*, mean speed, mean formation error, and acceleration cost.

Scen.	Strategy	Success rate	Mean *order* Φ	Mean speed (m/s) ($v_{\text{ref}} = 1$ m/s)	Mean formation error ε (m)	Acceleration cost (m^2/s^3)
1	BDC	**10/10**	0.9890	1.0249	0.3048	69.6589
	MPFC	8/10	**0.9934**	1.0639	0.6376	**23.7442**
	MPPDC	**10/10**	0.9824	**0.9863**	**0.2423**	25.0894
2	BDC	6/10	0.9883	**0.9887**	0.6872	53.8718
	MPFC	0/10	**0.9953**	1.0470	0.4593	**19.0365**
	MPPDC	**9/10**	0.9830	0.9800	**0.3217**	21.8559

increases, leading to an increased κ value, guiding the formation shape to return to its original shape.

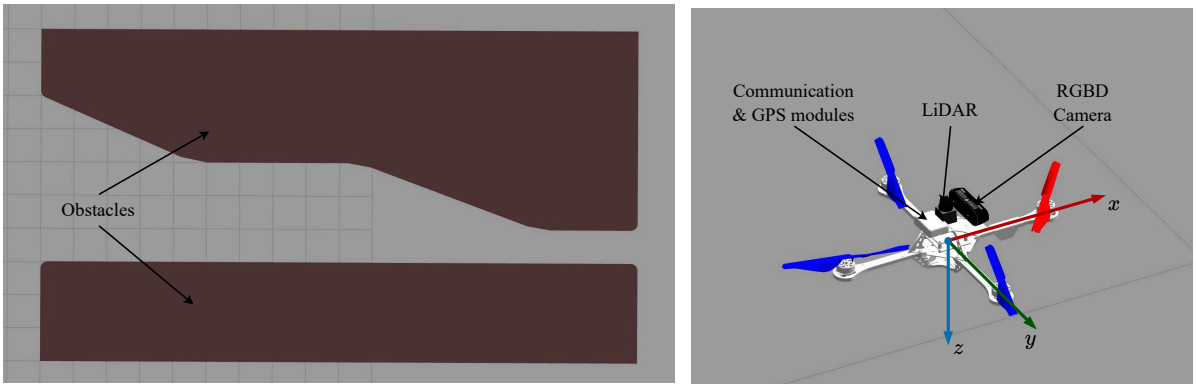
To describe the effectiveness of the MPPDC strategy, Fig. 4.5 illustrates the comparisons of our proposed strategy with other methods including speed profile, *order* metric, and *formation error*.

Fig. 4.5a-4.5b show the speed profiles of three strategies in both scenarios. The proposed strategy has demonstrated the ability to maintain a steady velocity around the desired velocity v_{ref}. The *BDC* shows the unstable speed when the formation changes its shape and/or navigates through the narrow passage due to the effect of the environment. Besides, *MPFC* indicates stability in achieving the desired velocity. However, when the robots in formation face a narrow environment, they do not maintain the designed velocity but are slightly larger.

Fig. 4.5c-4.5d present the *order* metric of three strategies. The results show the high consensus of robots in formation, represented by *order* value Φ approaching 1. Our *MPPDC* algorithm has demonstrated the ability to navigate formations more stably when encountering changes in narrow environments.

Fig. 4.5e-4.5f depict the superiority of the proposed *MPPDC* algorithm compared to other strategies in terms of formation maintenance. The *BDC* and *MPPDC* deformation strategies both present well-designed formation maintenance, as evidenced by the formation error converging to zero during movement. It can be seen that the proposed *MPPDC* controller maintains the designed formation better than *BDC*. Meanwhile, the configuration error of *MPFC* is very significant, because of the influence of obstacles on the ability to operate.

Table 4.1 presents a detailed comparison between the *BDC*, *MPFC*, and the proposed *MPPDC* strategies. The results demonstrate that the proposed *MPPDC* strategy outperforms *MPFC* in both scenarios and surpasses *BDC* in Scenario 2. In Scenario 1, the width-varying tunnel allows the *BDC* to switch modes and configurations smoothly and efficiently. However, in Scenario 2, where the tunnel's width changes suddenly, the formation struggles to adapt its configuration in time. The proposed *MPPDC* excels in maintaining the formation due to its minimal formation error, ε.



(a) The cave-like environment (b) The used drone model [59], [60]

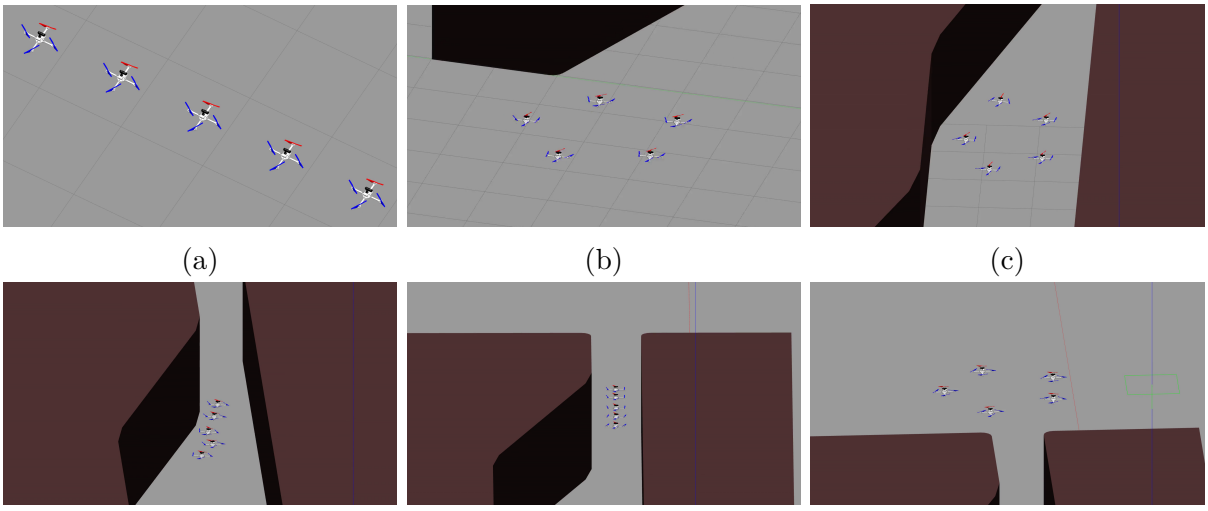
Figure 4.7: The SIL set up on Gazebo simulator. The environment consists of two large obstacles forming a cave-like environment. We use 5 Hummingbird drones equipped with perception sensors including LiDAR, RGBD camera, communication and positioning modules.

While the *MPFC* exhibits the best acceleration cost-owing to the absence of transformations between modes-the cost associated with the *MPPDC* is slightly higher but still shows significant improvement over *BDC*. Both deformation control methods (*MPFC* and *MPPDC*) demonstrate effective speed maintenance, as they approximately sustain the desired reference speed, v_{ref}. The mean order in all three strategies is close to 1, reaffirming their efficiency in maintaining the formation.

We further investigate the effect of swarm size on the performance of the proposed strategy by using swarms of 4, 5, 7, 10, 15, and 20 robots. We evaluate the changes in the average *order* value Φ and formation error ε during the robot swarm's movement, as illustrated in Fig. 4.6. The results indicate that as the number of robots in the formation increases, the proposed strategy maintains a high level of navigation ability, with the average *order* value remaining close to 1. When considering the influence of the number of robots on the average formation error ε value, we exclude the formation generation stage and the transition stage. The results show that, as the number of robots in the formation increases, the formation error remains relatively stable, fluctuating around 0.1 m. Additionally, statistics show that the time required to generate a formation from the initial position, t_1, and the time needed to switch formations between different modes, t_2, both increase as the number of robots in the formation grows. Specifically, for a formation of 4 robots, $t_1 \approx 2.5$ s and $t_2 \approx 2$ s, for a formation of 20 robots, $t_1 \approx 6.5$ s and $t_2 \approx 5.2$ s. Overall, the *MPPDC* algorithm has been shown to be effectively deployable in scalable robot swarms, confirming its robustness and efficiency across various swarm sizes.

4.4.2 Software-in-the-loop verification

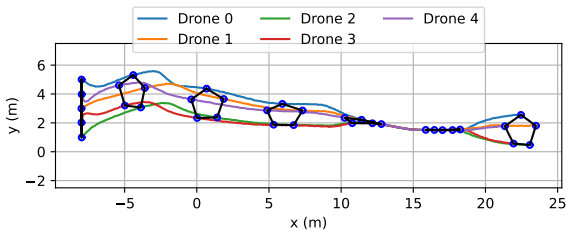
For the further performance assessment of the proposed control strategy, we have carried out software-in-the-loop (SIL) tests, as illustrated in Fig. 4.7, whose environment is a narrow space that consists of two large obstacles form a cave-like environment with different widths in each area in Gazebo simulator, as shown in Fig. 4.7a. We set up five



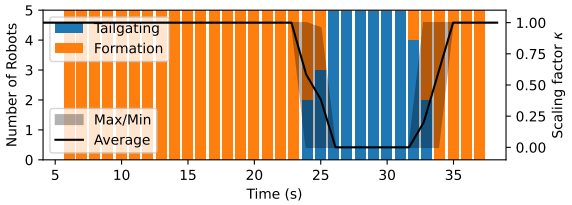
(a) (b) (c)

(d) (e) (f)

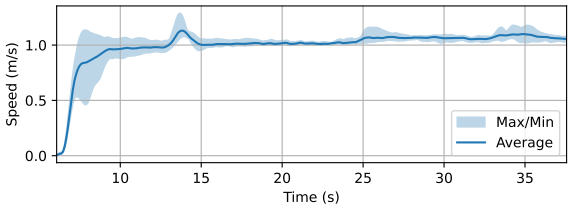
Figure 4.8: Snapshot of the migration process of a formation using the proposed strategy. At the beginning of the SIL test, drones in a formation hover and start at their initial positions (a). Consequently, they converge to the specified shape (b), i.e. pentagon shape. When they sense that the width of the environment is not enough to maintain their original shape, they deform their shape to adapt to the environment (c). When the formation finds that the environment in the front is too narrow, the formation transforms into a straight line shape (d) and moves safely through the narrow passage (e). Finally, after escaping from the narrow passage, they transform back to their original shape (f) and reach the target.



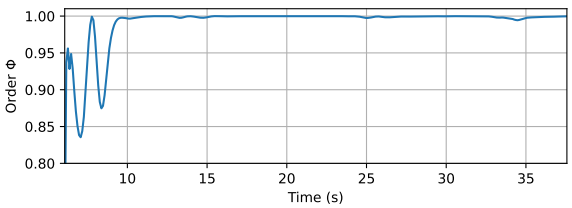
(a) The captured motion paths



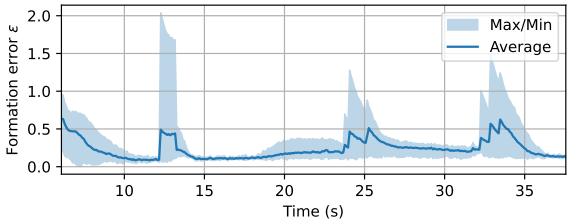
(b) Correlation of number of robots and scaling factor κ



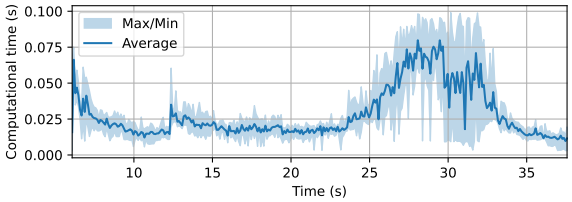
(c) The speed profile



(d) The *order* metric Φ



(e) The *formation error* ε_i



(f) The computational times

Figure 4.9: The recorded results and metrics in the SIL test

homogeneous Hummingbird quadrotors[1] developed based on the RotorS simulator [59] with an arm length of 0.17 m and a mass of 0.716 kg, the rotor thrust constant is 1.6^{-2} N/A and the rotor drag constant is 8.54858×10^{-6} Nm/A, as depicted in Fig. 4.7b. Each UAV is equipped with local sensors that provide point cloud data of the environment, a positioning module that supplies location information, and a communication module for interacting with other UAVs.

Fig. 4.8 depicts the migration process of a formation passing through the narrow space. It can be seen that a formation can successfully move through the tight passage without collision. During the movement process, a formation can adapt its shape based on the width of passages, which is measured and estimated by each UAV, which can be described in Fig. 4.9a. Fig. 4.9 provides in detail the results of the SIL test. According to Fig. 4.9b, each UAV can self-assign its mode based on the perception of the surrounding environment. The formation shrinks as the environment shrinks, described by the κ value, as illustrated in Fig. 4.9b. Drones in the formation transform to the *"Tailgating"* mode when the narrow space is detected. The requirements for the movement mission on the speed, order, and formation keeping are also guaranteed, as depicted in Fig. 4.9c-4.9d-4.9e. Moreover, the computational time to handle each iteration of the controller can be reported in Fig. 4.9f. It can be confirmed that the performance of the proposed control can be successfully deployed for the system.

4.4.3 Discussion

Through the results of simulation experiments, we show the key properties of the proposed strategy as follows:

Decentralization

MPPDC is a fully decentralized control. Each robot's decisions are based solely on its own local sensor perceptions. Unlike the approach in [11] which requires network-wide communication under limited communication to update the information of all robots, the *MPPDC* utilizes only one-hop communication between each robot and its neighbors to update predictive states.

High performance

The simulation results in 4.4.1 show that *MPPDC* successfully navigates robots through narrow environments, ensuring high-performance indicators such as low formation error, stable formation direction, and consistent speed. In contrast to the studies in [10]–[12], [64] which only shrink or expand the formation to adapt to changing space and have no evidence of movement through very narrow spaces, *MPPDC* enables the robot formation to transition into a tailgating (one-chain) configuration, allowing the robots to easily move through the very narrow spaces one at a time.

[1]Source code used to setup Gazebo SIL test - https://github.com/duynamrcv/hummingbird_simulator

Scalability

The MPPDC algorithm is successfully implemented with variable-sized robot swarms of 4, 5, 6, 7, 10, 15, and 20 robots without requiring any modifications to the algorithm. The measurement metrics of the swarm indicate that the requirements are satisfied, proving that MPPDC is scalable and maintains efficiency across different swarm sizes.

Flexibility

The *MPPDC* algorithm provides flexibility and adaptability to different environmental geometries, automatically adjusting structures based on the scaling factor κ in Eq. (4.22) obtained from local sensor perceptual information. By analyzing and processing data from local sensors equipped on the robot, the swarm can adapt to numerous environmental structures. In narrow spaces, the proposed algorithm provides each robot with the ability for distributed decision-making, i.e., the formation can shrink/expand or transform to a straight line, to navigate effectively and safely. MPPDC was also investigated on a simulation platform that replicates the physical environment, demonstrating its potential for applications in real-world scenarios.

4.5 Conclusion

In this paper, we have presented an optimal model prediction-based perceptual deformation control strategy to navigate the formation of robots through narrow environments. The proposed strategy enables each robot to be able to collect point cloud data to estimate the width of environments and perform distributed decision-making to switch between modes. This approach allows the formation to adapt its shape to the available space or to transform into a straight-line formation to pass through the narrow passages. The movement requirements, constraints, and limitations of the formation are modeled using a weighted-sum cost function to generate the optimal control signal. Through the simulations and comparisons, the performance of the proposed method can be confirmed.

Bibliography

[1] P. Shi and B. Yan, "A survey on intelligent control for multiagent systems," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 1, pp. 161–175, 2021. DOI: 10.1109/TSMC.2020.3042823.

[2] G. Skorobogatov, C. Barrado, and E. Salamí, "Multiple uav systems: A survey," *Unmanned Systems*, vol. 08, no. 02, pp. 149–169, Apr. 2020, ISSN: 2301-3869. DOI: 10.1142/s2301385020500090.

[3] J. Tang, H. Duan, and S. Lao, "Swarm intelligence algorithms for multiple unmanned aerial vehicles collaboration: A comprehensive review," *Artificial Intelligence Review*, vol. 56, no. 5, pp. 4295–4327, Sep. 2022, ISSN: 1573-7462. DOI: 10.1007/s10462-022-10281-7.

[4] K.-K. Oh, M.-C. Park, and H.-S. Ahn, "A survey of multi-agent formation control," *Automatica*, vol. 53, pp. 424–440, Mar. 2015, ISSN: 0005-1098. DOI: 10.1016/j.automatica.2014.10.022.

[5] S. Huang, R. S. H. Teo, and K. K. Tan, "Collision avoidance of multi unmanned aerial vehicles: A review," *Annual Reviews in Control*, vol. 48, pp. 147–164, 2019, ISSN: 1367-5788. DOI: 10.1016/j.arcontrol.2019.10.001.

[6] H. Rastgoftar and E. M. Atkins, "Safe multi-cluster uav continuum deformation coordination," *Aerospace Science and Technology*, vol. 91, Aug. 2019, ISSN: 1270-9638. DOI: 10.1016/j.ast.2019.05.002.

[7] T. Balch and R. Arkin, "Behavior-based formation control for multirobot teams," *IEEE Transactions on Robotics and Automation*, vol. 14, no. 6, pp. 926–939, 1998. DOI: 10.1109/70.736776.

[8] F. Berlinger, M. Gauci, and R. Nagpal, "Implicit coordination for 3D underwater collective behaviors in a fish-inspired robot swarm," *Science Robotics*, vol. 6, no. 50, Jan. 2021, ISSN: 2470-9476. DOI: 10.1126/scirobotics.abd8668.

[9] T. Nhu, P. D. Hung, V. A. Ho, and T. D. Ngo, "Fuzzy-based distributed behavioral control with wall-following strategy for swarm navigation in arbitrary-shaped environments," *IEEE Access*, vol. 9, pp. 139 176–139 185, 2021. DOI: 10.1109/ACCESS.2021.3119232.

[10] B. G. Elkilany, A. A. Abouelsoud, A. M. R. Fathelbab, and H. Ishii, "A proposed decentralized formation control algorithm for robot swarm based on an optimized potential field method," *Neural Computing and Applications*, vol. 33, no. 1, pp. 487–499, May 2020, ISSN: 1433-3058. DOI: 10.1007/s00521-020-05032-0.

[11] J. Alonso-Mora, E. Montijano, T. Nägeli, O. Hilliges, M. Schwager, and D. Rus, "Distributed multi-robot formation control in dynamic environments," *Autonomous Robots*, vol. 43, no. 5, pp. 1079–1100, Jul. 2018, ISSN: 1573-7527. DOI: 10.1007/s10514-018-9783-9.

[12] G. Vásárhelyi, C. Virágh, G. Somorjai, T. Nepusz, A. E. Eiben, and T. Vicsek, "Optimized flocking of autonomous drones in confined environments," *Science Robotics*, vol. 3, no. 20, Jul. 2018, ISSN: 2470-9476. DOI: 10.1126/scirobotics.aat3536.

[13] D. Zhou, Z. Wang, S. Bandyopadhyay, and M. Schwager, "Fast, on-line collision avoidance for dynamic vehicles using buffered voronoi cells," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 1047–1054, 2017. DOI: 10.1109/LRA.2017.2656241.

[14] Y. Wu, J. Gou, X. Hu, and Y. Huang, "A new consensus theory-based method for formation control and obstacle avoidance of UAVs," *Aerospace Science and Technology*, vol. 107, p. 106 332, Dec. 2020, ISSN: 1270-9638. DOI: 10.1016/j.ast.2020.106332.

[15] Y. Q. Chen and Z. Wang, "Formation control: A review and a new consideration," in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2005, pp. 3181–3186. DOI: 10.1109/IROS.2005.1545539.

[16] A. S. Brandão and M. Sarcinelli-Filho, "On the guidance of multiple uav using a centralized formation control scheme and delaunay triangulation," *Journal of Intelligent & Robotic Systems*, vol. 84, no. 1–4, pp. 397–413, Nov. 2015, ISSN: 1573-0409. DOI: 10.1007/s10846-015-0300-5.

[17] Y. Liu and R. Bucknall, "A survey of formation control and motion planning of multiple unmanned vehicles," *Robotica*, vol. 36, no. 7, pp. 1019–1047, Mar. 2018, ISSN: 1469-8668. DOI: 10.1017/s0263574718000218.

[18] H.-S. Ahn, *Formation Control: Approaches for Distributed Agents*. Springer International Publishing, 2020, ISBN: 9783030151874. DOI: 10.1007/978-3-030-15187-4.

[19] J. Hu, H. Zhang, L. Liu, X. Zhu, C. Zhao, and Q. Pan, "Convergent multiagent formation control with collision avoidance," *IEEE Transactions on Robotics*, vol. 36, no. 6, pp. 1805–1818, 2020. DOI: 10.1109/TRO.2020.2998766.

[20] F. Liao, R. Teo, J. L. Wang, X. Dong, F. Lin, and K. Peng, "Distributed formation and reconfiguration control of vtol uavs," *IEEE Transactions on Control Systems Technology*, vol. 25, no. 1, pp. 270–277, 2017. DOI: 10.1109/TCST.2016.2547952.

[21] A. D. Dang, H. M. La, T. Nguyen, and J. Horn, "Formation control for autonomous robots with collision and obstacle avoidance using a rotational and repulsive force–based approach," *International Journal of Advanced Robotic Systems*, vol. 16, no. 3, May 2019. DOI: 10.1177/1729881419847897.

[22] A. Mirzaeinia, M. Hassanalian, K. Lee, and M. Mirzaeinia, "Energy conservation of v-shaped swarming fixed-wing drones through position reconfiguration," *Aerospace Science and Technology*, vol. 94, p. 105 398, Nov. 2019. DOI: 10.1016/j.ast.2019.105398.

[23] H. Zhu, J. Juhl, L. Ferranti, and J. Alonso-Mora, "Distributed multi-robot formation splitting and merging in dynamic environments," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 9080–9086. DOI: 10.1109/ICRA.2019.8793765.

[24] H. P. Quang, T. Nguyen Dam, V. N. Hoang, and H. Pham Duy, "Multi-UAV coverage strategy with v-shaped formation for patrol and surveillance," in *2022 11th International Conference on Control, Automation and Information Sciences (ICCAIS)*, 2022, pp. 487–492. DOI: 10.1109/ICCAIS56082.2022.9990236.

[25] D. Zhang and H. Duan, "Switching topology approach for uav formation based on binary-tree network," *Journal of the Franklin Institute*, vol. 356, no. 2, pp. 835–859, Jan. 2019, ISSN: 0016-0032. DOI: 10.1016/j.jfranklin.2017.11.026.

[26] H. Shakhatreh, A. H. Sawalmeh, A. Al-Fuqaha, *et al.*, "Unmanned aerial vehicles (UAVs): A survey on civil applications and key research challenges," *IEEE Access*, vol. 7, pp. 48 572–48 634, 2019. DOI: 10.1109/ACCESS.2019.2909530.

[27] N. Duong Thi Thuy, D. Nam Bui, M. Duong Phung, and H. Pham Duy, "Deployment of UAVs for optimal multihop ad-hoc networks using particle swarm optimization and behavior-based control," in *2022 11th International Conference on Control, Automation and Information Sciences (ICCAIS)*, 2022, pp. 304–309. DOI: 10.1109/ICCAIS56082.2022.9990164.

[28] B. D. O. Anderson, B. Fidan, C. Yu, and D. Walle, "UAV formation control: Theory and application," in *Lecture Notes in Control and Information Sciences*, Springer London, 2008, pp. 15–33. DOI: 10.1007/978-1-84800-155-8_2.

[29] T. Balch, "Hierarchic social entropy: An information theoretic measure of robot group diversity," *Autonomous Robots*, vol. 8, no. 3, pp. 209–238, 2000. DOI: 10.1023/a:1008973424594.

[30] J. Alonso-Mora, E. Montijano, M. Schwager, and D. Rus, "Distributed multi-robot formation control among obstacles: A geometric and optimization approach with consensus," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 5356–5363. DOI: 10.1109/ICRA.2016.7487747.

[31] V. Hoang, M. Phung, T. Dinh, Q. Zhu, and Q. Ha, "Reconfigurable multi-UAV formation using angle-encoded pso," in *2019 IEEE 15th International Conference on Automation Science and Engineering (CASE)*, 2019, pp. 1670–1675. DOI: 10.1109/COASE.2019.8843165.

[32] D. Roy, A. Chowdhury, M. Maitra, and S. Bhattacharya, "Multi-robot virtual structure switching and formation changing strategy in an unknown occluded environment," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 4854–4861. DOI: 10.1109/IROS.2018.8594438.

[33] Q. Feng, X. Hai, B. Sun, *et al.*, "Resilience optimization for multi-UAV formation reconfiguration via enhanced pigeon-inspired optimization," *Chinese Journal of Aeronautics*, vol. 35, no. 1, pp. 110–123, Jan. 2022. DOI: 10.1016/j.cja.2020.10.029.

[34] C. Gao, J. Ma, T. Li, and Y. Shen, "Hybrid swarm intelligent algorithm for multi-UAV formation reconfiguration," *Complex & Intelligent Systems*, vol. 9, no. 2, pp. 1929–1962, Oct. 2022. DOI: 10.1007/s40747-022-00891-7.

[35] M. J. Matarić and F. Michaud, "Behavior-based systems," in *Springer Handbook of Robotics*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2008, pp. 891–909, ISBN: 978-3-540-30301-5. DOI: 10.1007/978-3-540-30301-5_39.

[36] A. M. Kahagh, F. Pazooki, and S. E. Haghighi, "Obstacle avoidance in v-shape formation flight of multiple fixed-wing UAVs using variable repulsive circles," *The Aeronautical Journal*, vol. 124, no. 1282, pp. 1979–2000, Oct. 2020. DOI: 10.1017/aer.2020.81.

[37] T. Vicsek, A. Czirók, E. Ben-Jacob, I. Cohen, and O. Shochet, "Novel type of phase transition in a system of self-driven particles," *Physical Review Letters*, vol. 75, no. 6, pp. 1226–1229, Aug. 1995. DOI: 10.1103/physrevlett.75.1226.

[38] Y. Cao, W. Yu, W. Ren, and G. Chen, "An overview of recent progress in the study of distributed multi-agent coordination," *IEEE Transactions on Industrial Informatics*, vol. 9, no. 1, pp. 427–438, 2013. DOI: 10.1109/TII.2012.2219061.

[39] A. Farinelli, L. Iocchi, and D. Nardi, "Multirobot systems: A classification focused on coordination," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 34, no. 5, pp. 2015–2028, 2004. DOI: 10.1109/TSMCB.2004.832155.

[40] X. Dong, B. Yu, Z. Shi, and Y. Zhong, "Time-varying formation control for unmanned aerial vehicles: Theories and applications," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 1, pp. 340–348, Jan. 2015, ISSN: 1558-0865. DOI: 10.1109/tcst.2014.2314460. [Online]. Available: http://dx.doi.org/10.1109/TCST.2014.2314460.

[41] X. Dong and G. Hu, "Time-varying formation control for general linear multi-agent systems with switching directed topologies," *Automatica*, vol. 73, pp. 47–55, Nov. 2016, ISSN: 0005-1098. DOI: 10.1016/j.automatica.2016.06.024. [Online]. Available: http://dx.doi.org/10.1016/j.automatica.2016.06.024.

[42] C. W. Reynolds, "Flocks, herds and schools: A distributed behavioral model," *ACM SIGGRAPH Computer Graphics*, vol. 21, no. 4, pp. 25–34, Aug. 1987, ISSN: 0097-8930. DOI: 10.1145/37402.37406.

[43] G. Antonelli, F. Arrichiello, and S. Chiaverini, "Flocking for multi-robot systems via the null-space-based behavioral control," *Swarm Intelligence*, vol. 4, no. 1, pp. 37–56, Oct. 2009, ISSN: 1935-3820. DOI: 10.1007/s11721-009-0036-6. [Online]. Available: http://dx.doi.org/10.1007/s11721-009-0036-6.

[44] S. Hauert, S. Leven, M. Varga, *et al.*, "Reynolds flocking in reality with fixed-wing robots: Communication range vs. maximum turning rate," in *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011, pp. 5015–5020. DOI: 10.1109/IROS.2011.6095129.

[45] 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. DOI: 10.1109/TAC.2005.864190.

[46] D. N. Bui, M. D. Phung, and H. P. Duy, "Self-reconfigurable V-shape formation of multiple UAVs in narrow space environments," in *2024 IEEE/SICE International Symposium on System Integration (SII)*, 2024, pp. 1006–1011. DOI: 10.1109/SII58957.2024.10417519.

[47] P. D. Hung, T. Q. Vinh, and T. D. Ngo, "Hierarchical distributed control for global network integrity preservation in multirobot systems," *IEEE Transactions on Cybernetics*, vol. 50, no. 3, pp. 1278–1291, 2020. DOI: 10.1109/TCYB.2019.2913326.

[48] P. Peng, W. Dong, G. Chen, and X. Zhu, "Obstacle avoidance of resilient uav swarm formation with active sensing system in the dense environment," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 10 529–10 535. DOI: 10.1109/IROS47612.2022.9981858.

[49] X. Zhou, J. Zhu, H. Zhou, C. Xu, and F. Gao, "Ego-swarm: A fully autonomous and decentralized quadrotor swarm system in cluttered environments," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 4101–4107. DOI: 10.1109/ICRA48506.2021.9561902.

[50] T. Dang, M. Tranzatto, S. Khattak, F. Mascarich, K. Alexis, and M. Hutter, "Graph-based subterranean exploration path planning using aerial and legged robots," *Journal of Field Robotics*, vol. 37, no. 8, pp. 1363–1388, Nov. 2020. DOI: 10.1002/rob.21993. [Online]. Available: https://doi.org/10.1002/rob.21993.

[51] J. P. Queralta, J. Taipalmaa, B. Can Pullinen, *et al.*, "Collaborative multi-robot search and rescue: Planning, coordination, perception, and active vision," *IEEE Access*, vol. 8, pp. 191 617–191 643, 2020. DOI: 10.1109/ACCESS.2020.3030190.

[52] M. Saska, D. Hert, T. Baca, V. Kratky, and T. Nascimento, "Formation control of unmanned micro aerial vehicles for straitened environments," *Autonomous Robots*, vol. 44, no. 6, pp. 991–1008, Mar. 2020. DOI: 10.1007/s10514-020-09913-0.

[53] J. V. Gómez, A. Lumbier, S. Garrido, and L. Moreno, "Planning robot formations with fast marching square including uncertainty conditions," *Robotics and Autonomous Systems*, vol. 61, no. 2, pp. 137–152, Feb. 2013, ISSN: 0921-8890. DOI: 10.1016/j.robot.2012.10.009. [Online]. Available: http://dx.doi.org/10.1016/j.robot.2012.10.009.

[54] H. Ebel, E. Sharafian Ardakani, and P. Eberhard, "Distributed model predictive formation control with discretization-free path planning for transporting a load," *Robotics and Autonomous Systems*, vol. 96, pp. 211–223, Oct. 2017, ISSN: 0921-8890. DOI: 10.1016/j.robot.2017.07.007. [Online]. Available: http://dx.doi.org/10.1016/j.robot.2017.07.007.

[55] D. Roy, A. Chowdhury, M. Maitra, and S. Bhattacharya, "Multi-robot virtual structure switching and formation changing strategy in an unknown occluded environment," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, IEEE, Oct. 2018. DOI: 10.1109/iros.2018.8594438. [Online]. Available: https://doi.org/10.1109/iros.2018.8594438.

[56] J. Alonso-Mora, S. Baker, and D. Rus, "Multi-robot formation control and object transport in dynamic environments via constrained optimization," *The International Journal of Robotics Research*, vol. 36, no. 9, pp. 1000–1021, Aug. 2017, ISSN: 1741-3176. DOI: 10.1177/0278364917719333. [Online]. Available: http://dx.doi.org/10.1177/0278364917719333.

[57] X. Fu, J. Pan, H. Wang, and X. Gao, "A formation maintenance and reconstruction method of UAV swarm based on distributed control," *Aerospace Science and Technology*, vol. 104, p. 105 981, Sep. 2020, ISSN: 1270-9638. DOI: 10.1016/j.ast.2020.105981.

[58] C. Bai, P. Yan, W. Pan, and J. Guo, "Learning-based multi-robot formation control with obstacle avoidance," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 8, pp. 11 811–11 822, 2022. DOI: 10.1109/TITS.2021.3107336.

[59] F. Furrer, M. Burri, M. Achtelik, and R. Siegwart, "Robot operating system (ros): The complete reference (volume 1)," in A. Koubaa, Ed. Cham: Springer International Publishing, 2016, ch. RotorS—A Modular Gazebo MAV Simulator Framework, pp. 595–625, ISBN: 978-3-319-26054-9. DOI: 10.1007/978-3-319-26054-9_23.

[60] D. N. Bui, T. T. Van Nguyen, and M. D. Phung, "Lyapunov-based nonlinear model predictive control for attitude trajectory tracking of unmanned aerial vehicles," *International Journal of Aeronautical and Space Sciences*, vol. 24, no. 2, pp. 502–513, Oct. 2022, ISSN: 2093-2480. DOI: 10.1007/s42405-022-00545-5.

[61] M. Nagy, Z. Ákos, D. Biro, and T. Vicsek, "Hierarchical group dynamics in pigeon flocks," *Nature*, vol. 464, no. 7290, pp. 890–893, Apr. 2010, ISSN: 1476-4687. DOI: 10.1038/nature08891.

[62] Y. Koren and J. Borenstein, "Potential field methods and their inherent limitations for mobile robot navigation," in *Proceedings. 1991 IEEE International Conference on Robotics and Automation*, 1991, 1398–1404 vol.2. DOI: 10.1109/ROBOT.1991.131810.

[63] L. E. Beaver and A. A. Malikopoulos, "An overview on optimal flocking," *Annual Reviews in Control*, vol. 51, pp. 88–99, 2021, ISSN: 1367-5788. DOI: 10.1016/j.arcontrol.2021.03.004.

[64] E. Soria, F. Schiano, and D. Floreano, "Predictive control of aerial swarms in cluttered environments," *Nature Machine Intelligence*, vol. 3, no. 6, pp. 545–554, May 2021, ISSN: 2522-5839. DOI: 10.1038/s42256-021-00341-y.

[65] C. E. Luis, M. Vukosavljev, and A. P. Schoellig, "Online trajectory generation with distributed model predictive control for multi-robot motion planning," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 604–611, 2020. DOI: 10.1109/LRA.2020.2964159.

[66] M. L. Darby and M. Nikolaou, "MPC: Current practice and challenges," *Control Engineering Practice*, vol. 20, no. 4, pp. 328–342, Apr. 2012, ISSN: 0967-0661. DOI: 10.1016/j.conengprac.2011.12.004.

[67] P. Virtanen, R. Gommers, T. E. Oliphant, *et al.*, "SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python," *Nature Methods*, vol. 17, pp. 261–272, 2020. DOI: 10.1038/s41592-019-0686-2.

[68] E. Soria, F. Schiano, and D. Floreano, "Distributed predictive drone swarms in cluttered environments," *IEEE Robotics and Automation Letters*, vol. 7, no. 1, pp. 73–80, 2022. DOI: 10.1109/LRA.2021.3118091.

[69] A. S. Matveev, V. V. Magerkin, and A. V. Savkin, "A method of reactive control for 3d navigation of a nonholonomic robot in tunnel-like environments," *Automatica*, vol. 114, p. 108 831, Apr. 2020, ISSN: 0005-1098. DOI: 10.1016/j.automatica.2020.108831. [Online]. Available: http://dx.doi.org/10.1016/j.automatica.2020.108831.

[70] X. Dong, B. Yu, Z. Shi, and Y. Zhong, "Time-varying formation control for unmanned aerial vehicles: Theories and applications," *IEEE Transactions on Control Systems Technology*, vol. 23, no. 1, pp. 340–348, 2015. DOI: 10.1109/TCST.2014.2314460.

[71] B. Lindqvist, S. S. Mansouri, A.-a. Agha-mohammadi, and G. Nikolakopoulos, "Nonlinear MPC for collision avoidance and control of UAVs with dynamic obstacles," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 6001–6008, 2020. DOI: 10.1109/LRA.2020.3010730.

[72] D. Saccani, L. Cecchin, and L. Fagiano, "Multitrajectory model predictive control for safe UAV navigation in an unknown environment," *IEEE Transactions on Control Systems Technology*, vol. 31, no. 5, pp. 1982–1997, 2023. DOI: 10.1109/TCST.2022.3216989.

[73] D. Fox, W. Burgard, and S. Thrun, "The dynamic window approach to collision avoidance," *IEEE Robotics & Automation Magazine*, vol. 4, no. 1, pp. 23–33, 1997. DOI: 10.1109/100.580977.

[74] M. Kamel, J. Alonso-Mora, R. Siegwart, and J. Nieto, "Robust collision avoidance for multiple micro aerial vehicles using nonlinear model predictive control," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 236–243. DOI: 10.1109/IROS.2017.8202163.

[75] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, ser. KDD'96, Portland, Oregon: AAAI Press, 1996, pp. 226–231.

[76] D. Kraft, *A Software Package for Sequential Quadratic Programming* (Deutsche Forschungs- und Versuchsanstalt für Luft- und Raumfahrt Köln: Forschungsbericht). Wiss. Berichtswesen d. DFVLR, 1988. [Online]. Available: https://books.google.com.vn/books?id=4rKaGwAACAAJ.

[77] J. Zhang, J. F. Campbell, D. C. Sweeney II, and A. C. Hupman, "Energy consumption models for delivery drones: A comparison and assessment," *Transportation Research Part D: Transport and Environment*, vol. 90, p. 102 668, Jan. 2021, ISSN: 1361-9209. DOI: 10.1016/j.trd.2020.102668.