

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**

Field: Electronics and Communications Engineering

Major: Electronics Engineering

Code: 8510302.01

**MASTER THESIS  
MAJOR: ELECTRONICS ENGINEERING**

Supervisor: **Dr. Pham Duy Hung**

Co-Supervisor: **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: .....

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: .....

Supervisor: **Dr. Pham Duy Hung**

Signature: .....

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

Signature: .....

---

# 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

<b>Acknowledgements</b>	iii
<b>Contents</b>	v
<b>List of Figures</b>	vii
<b>List of Tables</b>	viii
<b>Abbreviations and Table of symbols</b>	ix
<b>Preface</b>	x
<b>Abstract</b>	xi
<b>Chapter 1. Introduction</b>	1
1.1 Motivation . . . . .	1
1.2 Approaches and Background . . . . .	2
1.3 Contributions . . . . .	3
1.3.1 Research Contributions . . . . .	4
1.3.2 List of Publications . . . . .	5
1.3.3 Open-source Software . . . . .	5
<b>Chapter 2. Self-reconfigurable V-shape Formation of Multiple UAVs in Narrow Space Environments</b>	7
2.1 Introduction . . . . .	7
2.2 V-Shape Formation Design . . . . .	8
2.2.1 UAV model . . . . .	9
2.2.2 V-shape formation . . . . .	9
2.3 Distributed Formation Control Strategy . . . . .	10
2.3.1 Formation maintenance strategy . . . . .	10
2.3.2 Self-reconfigurable formation strategy . . . . .	12
2.3.3 Overall strategy . . . . .	12
2.4 Results and Discussion . . . . .	13
2.4.1 Simulation setup . . . . .	13
2.4.2 Results . . . . .	13
2.5 Conclusion . . . . .	15

<b>Chapter 3. Event-based Reconfiguration Control for Time-varying Formation of Robot Swarms in Narrow Spaces</b>	16
3.1 Introduction . . . . .	16
3.2 Preliminaries . . . . .	18
3.2.1 Robot Model . . . . .	18
3.2.2 Problem formulation . . . . .	20
3.2.3 Formation configurations . . . . .	20
3.3 Event-based Reconfiguration Control . . . . .	21
3.3.1 Individual behaviors . . . . .	21
3.3.2 Event-based Reconfiguration Control strategy . . . . .	23
3.3.3 Stability analysis . . . . .	24
3.4 Results . . . . .	26
3.4.1 Simulation and Comparison . . . . .	26
3.4.2 Validation on the software-in-the-loop Gazebo . . . . .	29
3.5 Conclusion . . . . .	31
<b>Chapter 4. Predictive Reconfiguration Control for Multi-Robot Formation in Cluttered Environments</b>	32
4.1 Introduction . . . . .	32
4.2 Formation Background . . . . .	34
4.3 Predictive Reconfiguration Control . . . . .	35
4.3.1 Predictive control design . . . . .	36
4.3.2 Formation reconfiguration . . . . .	39
4.4 Results and Discussion . . . . .	40
4.4.1 Evaluation Setup . . . . .	40
4.4.2 Results . . . . .	41
4.4.3 Software-in-the-loop verification . . . . .	43
4.4.4 Discussion . . . . .	45
4.5 Conclusion . . . . .	46
<b>Chapter 5. Conclusion and Future Works</b>	47
5.1 Conclusion . . . . .	47
5.2 Future direction . . . . .	48
<b>List of related publications</b>	49
<b>Bibliography</b>	50

---

# List of Figures

2.1	The self-reconfigurable V-shape formation can adjust its shape and navigate through narrow passages. . . . .	8
2.2	The sensing range $r_s$ and alert range $r_a$ ( $r_a < r_s$ ) of a UAV $R_i$ . . . . .	9
2.3	Illustration of the V-shape formation . . . . .	9
2.4	Self-reconfiguration of the V-shape formation based on the mechanism of pliers or scissors . . . . .	10
2.5	Simulation result of the V-shape formation moving through a narrow passage	13
2.6	Evaluation of the proposed algorithm . . . . .	14
3.1	We develop an event-based reconfiguration control (ERC) method to guide a TVF through narrow spaces. <i>Left:</i> Motion path of the TVF using purely behavior-based formation control [7], [12], which is collision with surrounding obstacles. <i>Right:</i> Motion path of the TVF using our proposed approach, which safely navigates through the narrow spaces. . . . .	17
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 white 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 $\mathcal{M}_i = \{\mathbf{o}\}$ (green) is the nearest point from robot $i$ to obstacle. . . . .	19
3.3	Schematic diagram of time-varying formation in the narrow spaces . . . . .	19
3.4	Overview of the proposed event-based reconfiguration control. The proposed strategy is constructed by two primary emergent strategy, including “ <i>formation</i> ” and “ <i>tailgating</i> ”, which are highlighted by red boxes. There are five individual behaviors that contribute to the emergent strategy, which illustrated via blue boxes. . . . .	21
3.5	Environment’s width estimation . . . . .	24
3.6	Motion paths and velocity profiles of the proposed ERC strategy in multiple configurations. . . . .	27
3.7	The <i>order</i> values of the proposed ERC strategy . . . . .	27
3.8	Correlation of robot number in each mode (bar chart) and scaling factor (black line) of the proposed ERC strategy . . . . .	27
3.9	Used Hummingbird UAV model . . . . .	29

3.10	The environment in SIL test . . . . .	30
3.11	Validation results captured in the SIL Gazebo . . . . .	30
3.12	The recorded paths of three UAV in the SIL test (Top view) . . . . .	31
4.1	Illustration of a robot with its range sensor having the scanning area $S_s$ (dashed gray circle) of radius $r_s$ and set $\mathcal{M}_i = \{\mathbf{m}\}$ (green) of the acquired point data. . . . .	35
4.2	Diagram of the proposed predictive reconfiguration control strategy. . . . .	35
4.3	The process of estimating the environment's width from the robot's range sensor. . . . .	37
4.4	Trajectories and formation shapes of the robots controlled by the PRC in two evaluating scenarios. . . . .	41
4.5	Comparison results of three control methods in two scenarios. . . . .	42
4.6	Effect of the swarm size on system performance, including the mean <i>order</i> $\Phi$ and formation error $\varepsilon$ . . . . .	42
4.7	The robot and environment structure used for software-in-the-loop tests. .	44
4.8	Reconfiguration process of the robot swarm in a SIL test: (a) initial positions of the robots; (b) form the desired pentagon shape; (c) shrink the formation in adaption to the environment; (d)-(e) switch to “ <i>tailgating</i> ” mode to travel through the narrow passage; (f) transform back to the desired shape. . . . .	44
4.9	Results of the SIL tests . . . . .	45

---

## List of Tables

2.1	Statistical evaluation of the proposed strategy for several different scenarios	13
3.1	Comparison between BC and our method, ERC. 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.	29
4.1	Comparison between BRC, PFC, and the proposed PRC	43

---

# Abbreviations and Table of symbols

## Abbreviations

APF	Artificial potential field
BC	Behavior-based Control
BRC	Behavior-based Reconfiguration Control
ERC	Event-based Reconfiguration Control
GPS	Global Positioning System
IMU	Inertial measurement unit
MPC	Model predictive control
NLP	Nonlinear programming
PFC	Predictive Formation Control
PRC	Predictive Reconfiguration Control
SAR	Search and Rescue
SLSQP	Sequential Least Squares Programming
TVF	Time-varying Formation
UAVs	Unmanned aerial vehicles

## Lists of Symbols

$r$	Robot radius	$r_a$	Alert radius
$r_s$	Sensing radius	$S_a$	Alert zone
$S_s$	Sensing zone	$\Psi$	Order
$\mathbf{p}$	Position	$\mathbf{v}$	Velocity
$\mathbf{u}$	Control signal	$\mathcal{N}$	Set of robots in formation
$\mathcal{N}_i$	Neighbor set of robot $i$	$\mathcal{M}$	Set of obstacle points
$\mathbf{u}_{\text{ref}}$	Desired direction	$v_{\text{ref}}$	Desired velocity
$d_{\text{ref}}$	Desired distance	$w_e$	Environmental width
$w_f$	Formation width	$\mathbf{L}$	Laplacian matrix
$\kappa$	Scaling factor	$P$	Horizontal length
$J$	Objective cost		

---

# Preface

This master's thesis is organized as a paper-based thesis with five chapters, with all relevant papers that constitute the research contributions included in the thesis. Specifically, Chapter 1 sets the stage for the dissertation by introducing the core issues addressed in this study. Chapter 2, chapter 3, and chapter 4 present the context of each method, corresponding to each article. Finally, Chapter 5 provides a summary of our findings and proposes potential future research directions to extend this work.

---

# 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 reconfiguration 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 reconfiguration control for multiple formation configurations is proposed. This method provides an effective strategy to transform the formation shape in narrow 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 narrow spaces. 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.

**Keywords:** unmanned aerial vehicles, multi-robot system, distributed control, formation control, reconfiguration control, artificial potential field, model predictive control

# 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 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 reconfiguration 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 reconfiguration 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 reconfiguration 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 for-

mation 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 reconfiguration control in multi-UAV systems.

This thesis aims to address the challenges of autonomous navigation by focusing on the coordination of multiple UAVs in reconfiguration 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 reconfiguration 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 interacts 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 narrow environments are quite limited.

To address these limitations in the context of V-shape formation movement through a constrained environment, in *Method 1*, 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 2* 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 3* 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 the master's thesis was to design control strategies to guide the formation of multiple robots through narrow 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 space. Aiming to develop an effective control strategy in the narrow 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 Reconfiguration Control for Time-varying Formation of Robot Swarms in Narrow Spaces

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 narrow space environments, this work proposed an event-based reconfiguration control strategy based on the artificial potential field (APF), which enhances the safety of multi-robot time-varying formation (TVF) in narrow spaces. 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 narrow spaces, which contributes to the configuration change if needed by applying the scale, rotation, or transformation into the straight line configuration to ensure collision-free motion.

#### **Method 3.** Predictive Reconfiguration Control for Multi-Robot Formation in Cluttered Environments

Aiming to improve the smoothness and optimize the motion control of multi-UAV formation given in *Method 2*, an optimal reconfiguration control strategy is presented for a decentralized multi-robot team to ensure safety in narrow space environments. The core contribution is in a perceptual reconfiguration 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, and Hung Pham Duy. “Event-based Reconfiguration Control for Time-varying Formation of Robot Swarms in Narrow Spaces,” *Manuscript*.
- **Duy-Nam Bui**, Manh Duong Phung, and Hung Pham Duy. “Predictive Reconfiguration Control for Multi-Robot Formation in Cluttered Environments,” *Manuscript*.

#### 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 (ICCR)*, 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](https://github.com/duynamrcv/hummingbird_simulator).

- The nonlinear model predictive control (MPC) for UAVs low-level controller [https://github.com/duynamrcv/hummingbird\\_mmpc](https://github.com/duynamrcv/hummingbird_mmpc).
- The self-reconfiguration strategy for V-shape formation of multiple UAVs: [https://github.com/duynamrcv/reconfigurable\\_vshape](https://github.com/duynamrcv/reconfigurable_vshape).
- The event-based reconfiguration control for multiple UAVs: <https://github.com/duynamrcv/erc>.
- The predictive reconfiguration control for multiple UAVs: <https://github.com/duynamrcv/prc>.

# 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

DOI: 10.1109/SII58957.2024.10417519

Code: [https://github.com/duynamrcv/reconfigurable\\_vshape](https://github.com/duynamrcv/reconfigurable_vshape)

Video: [https://youtu.be/\\_6u7yMNOySc](https://youtu.be/_6u7yMNOySc)

The selection of the V-shape formation is motivated by its maneuverability and visibility advantages. In this chapter, 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.

### 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 strategies 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.

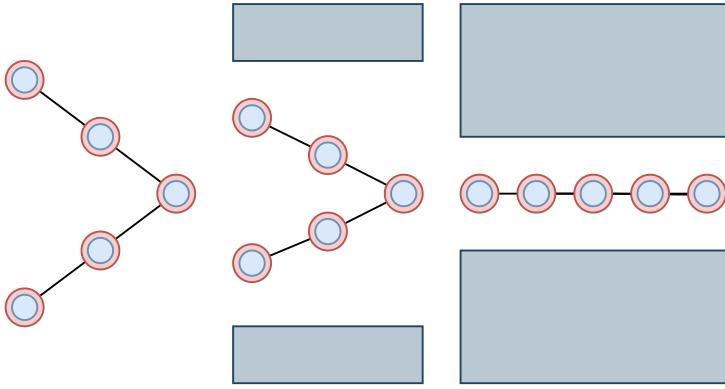


Figure 2.1: The self-reconfigurable V-shape formation can adjust its shape and navigate through narrow passages.

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 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.

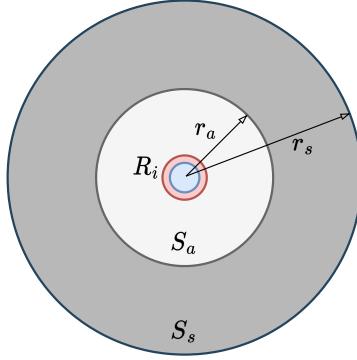


Figure 2.2: The sensing range  $r_s$  and alert range  $r_a$  ( $r_a < r_s$ ) of a UAV  $R_i$

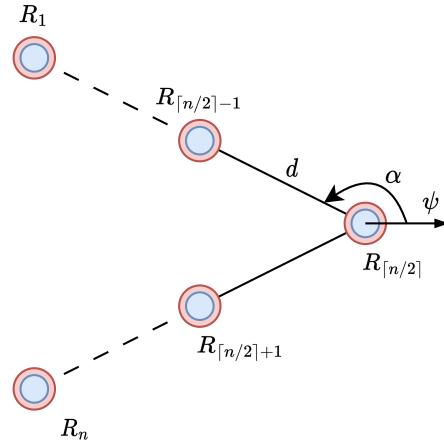


Figure 2.3: Illustration of the V-shape formation

### 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  $\mathbf{p}_i$  and heading angle  $\psi_i$ . The single-integrator kinematic model of UAV  $R_i$  can be expressed as follows:

$$\dot{\mathbf{p}}_i = \mathbf{v}_i, \quad (2.1)$$

where  $\mathbf{v}_i = [v_{ix}, v_{iy}]^T$  is the velocity vector of UAV  $R_i$ . The heading angle  $\psi_i$  then can be obtained as:

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

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,

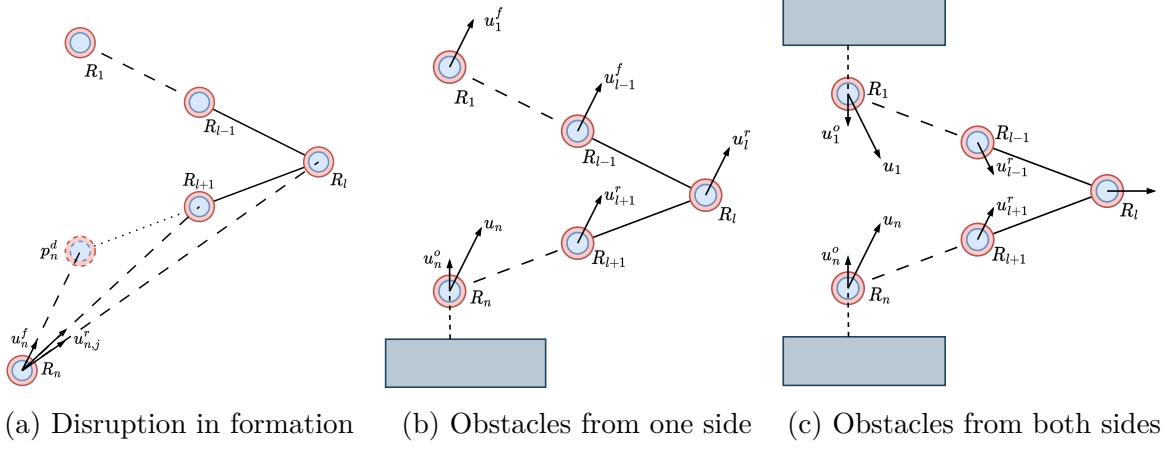


Figure 2.4: Self-reconfiguration of the V-shape formation based on the mechanism of pliers or scissors

$d$ , and the bearing angle between the formation heading and each wing,  $\alpha$ . Without loss of generality, choose UAV  $R_l$ , with  $l = \lceil n/2 \rceil$ , as the leader UAV located at the forefront of the formation. The desired distance  $d_i$  and angle  $\alpha_i$  between UAV  $R_i$ ,  $i \neq l$ , and UAV  $R_l$  are determined as follows:

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

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

$$\mathbf{p}_i^d = \mathbf{p}_l + d_i \begin{bmatrix} \cos \alpha_i \\ \sin \alpha_i \end{bmatrix}. \quad (2.4)$$

## 2.3 Distributed Formation Control Strategy

The formation control strategy aims to form, maintain and self-reconfigure 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 reconfiguring 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  $\mathbf{p}_i^d$  of UAV  $R_i$  in the formation can be obtained. Inspired by [21], [36], we define the formation behavior as follows:

$$\mathbf{v}_i^f = -k_f (\mathbf{p}_i - \mathbf{p}_i^d) + \mathbf{v}_l, \quad (2.5)$$

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  $\mathbf{p}_g$  be the goal position that the formation needs to reach. The goal reaching behavior is constructed as follows:

$$\mathbf{v}_i^g = -k_g (\mathbf{p}_i - \mathbf{p}_g), \quad (2.6)$$

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

## Obstacle avoidance behavior

During operation, the formation must avoid obstacles present in the environment. Let  $\mathbf{p}_{io_h}$  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:

$$\mathbf{v}_{ih}^o = \begin{cases} -k_o \left( \frac{1}{d_{io_h}^2} - \frac{1}{r_s^2} \right) \frac{\mathbf{p}_i - \mathbf{p}_{io_h}}{\|\mathbf{p}_i - \mathbf{p}_{io_h}\|}, & \text{if } d_{io_h} < r_s \\ 0, & \text{otherwise} \end{cases} \quad (2.7)$$

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:

$$\mathbf{v}_i^o = \sum_{h=1}^m \mathbf{v}_{ih}^o, \quad (2.8)$$

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

## 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.,  $\|\mathbf{p}_{ij}\| < r_s$ , will exert a repulsive force to preventing the UAVs from entering the alert area  $S_a$ . Let  $\mathbf{p}_{ij} = \mathbf{p}_i - \mathbf{p}_j$ . The collision avoidance behavior is determined as follows:

$$\mathbf{v}_{ij}^c = k_c \frac{e^{-\beta_c(\|\mathbf{p}_{ij}\|-r_a)}}{\|\mathbf{p}_{ij}\| - r_a} \frac{\mathbf{p}_i - \mathbf{p}_j}{\|\mathbf{p}_i - \mathbf{p}_j\|}, \quad (2.9)$$

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  $\mathbf{v}_n^o$  to avoid a potential collision with obstacles thus resulting in the control signal  $\mathbf{v}_n$ . Accordingly, other UAVs on the same V-wing, including the leader UAV, also adjust their position based on the behavior control signal  $\mathbf{v}_i^r$ . As the leader UAV  $R_l$  changes its position, the UAVs on the opposing wing realign themselves by formation behavior  $\mathbf{v}_i^f$ . 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 avoidance behaviors denoted as  $\mathbf{v}_1^o$  and  $\mathbf{v}_n^o$ . Other UAVs on the same wing respond by generating reconfiguration behaviors  $\mathbf{v}_i^r$  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:

$$\mathbf{v}_{ij}^r = k_r \frac{\|\mathbf{p}_{ij}\| - d_{ij}|^{\beta_r}}{(\|\mathbf{p}_{ij}\| - r_a)^2} \frac{\mathbf{p}_i - \mathbf{p}_j}{\|\mathbf{p}_i - \mathbf{p}_j\|}, \quad (2.10)$$

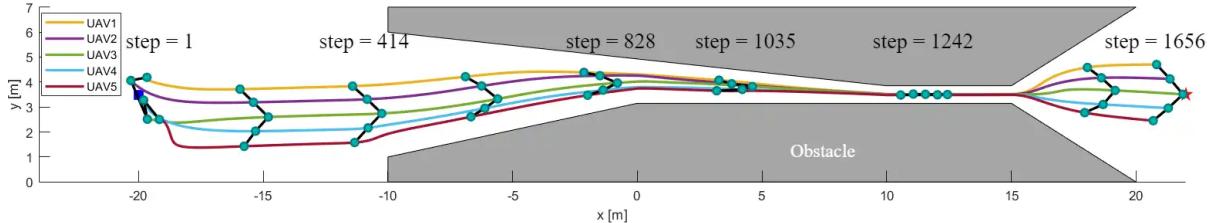
where  $k_r > 0$  is a positive reconfiguration gain,  $\beta_r > 0$  is the smoothness factor,  $d_{ij}$  is the desired distance between two UAVs  $R_i$  and  $R_j$  in the same wing,  $d_{ij} = d|i - j|$ . In (2.10), the term  $|\|\mathbf{p}_{ij}\| - d_{ij}|$  enables the UAVs to adjust their positions 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

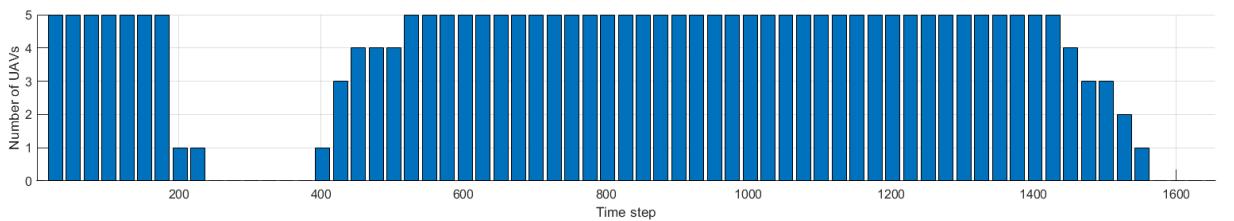
The overall distributed control strategy is obtained by combining the behaviors from all UAVs as follows:

$$\mathbf{v}_i = \begin{cases} \mathbf{v}_i^g + \mathbf{v}_i^r + \mathbf{v}_i^c + \mathbf{v}_i^o, & \text{if assigned as leader} \\ \mathbf{v}_i^f + \mathbf{v}_i^r + \mathbf{v}_i^c + \mathbf{v}_i^o. & \text{otherwise} \end{cases} \quad (2.11)$$

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.



(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

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

Scen.	Number UAVs	$d$	$\alpha$	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

## 2.4 Results and Discussion

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

### 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.

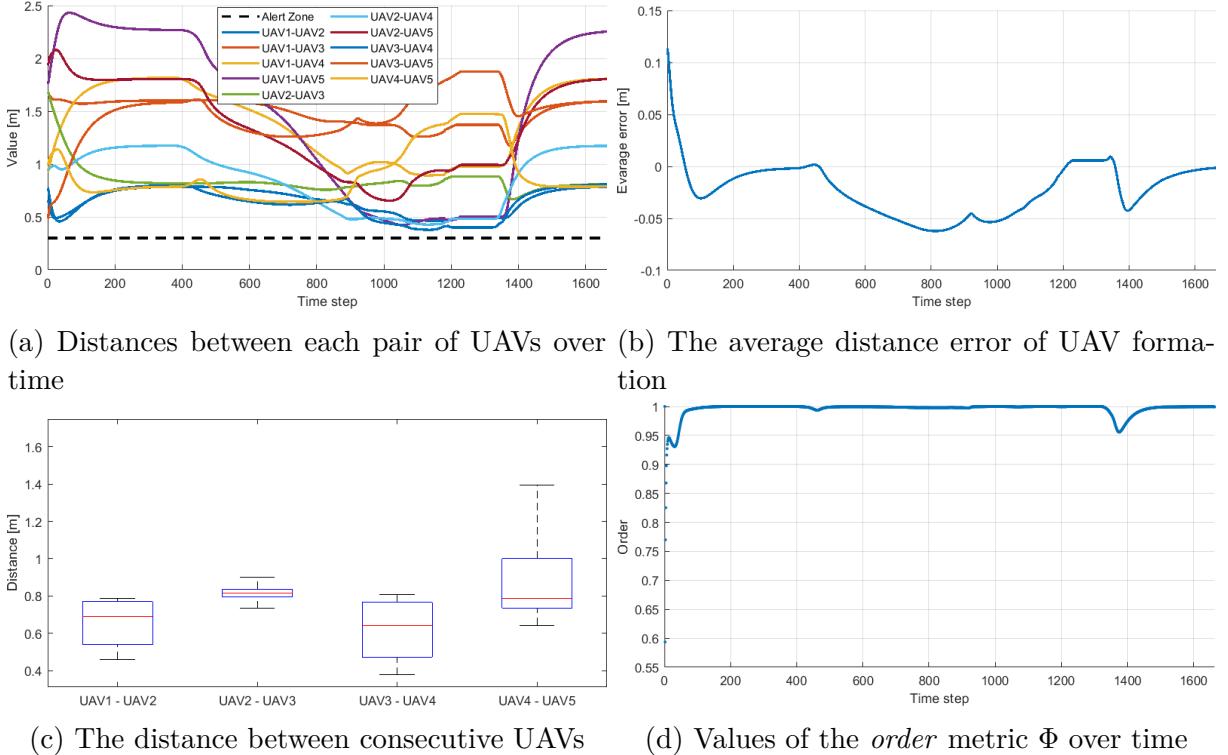


Figure 2.6: Evaluation of the proposed algorithm

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 the footnote<sup>1</sup>.

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 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  $\Phi$  that measures the similarity in the UAVs' direction is used [37]. It takes the values in range

<sup>1</sup>Simulation video: [https://youtu.be/\\_6u7yMNOySc](https://youtu.be/_6u7yMNOySc)

$[0, 1]$  and is computed as follows:

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

According to (2.12), the order metric  $\Phi$  is close to 1 when all UAVs have the same heading angle. Figure 2.6d shows the value of  $\Phi$  in our simulation. It can be seen that  $\Phi$  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,  $\alpha$ , 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 0.1 m, 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 Reconfiguration Control for Time-varying Formation of Robot Swarms in Narrow Spaces

*Manuscript*

Code: <https://github.com/duynamrcv/erc>

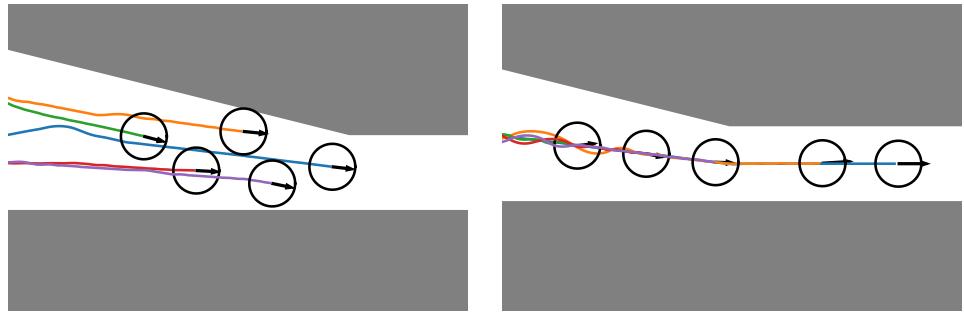
Video: <https://youtu.be/rCIjgSqiWXg>

This chapter aims to extend the previous method presented in Chapter 2 for multiple formation configurations in response to narrow spaces where the risk of collisions is increased due to limited spaces. An event-based reconfiguration control (ERC) is presented based on an artificial potential field (APF) for navigating a time-varying robot formation (TVF) through narrow spaces with multiple environmental types, including forest-like and tunnel-like environments.

### 3.1 Introduction

With advancements in networked multi-agent technology, multi-robot systems (MRSs) have rapidly developed towards autonomy, offering various applications from warehouse automation to search and rescue operations [38], [39]. A core element of these systems is a formation controller that enables the robots to collaborate in a desired configuration [4], [15]. In rigid formation control, achieving the desired configuration involves setting a specific target distance for each swarm agent. However, with the increasing complexity of formation tasks, the formation configuration needs to be adjustable to meet specific task requirements. Time-varying formation (TVF) control thus has become essential for swarm robots [40], [41].

Studies in [7], [42], [43] show that biological swarm motion can be characterized by three primary behavioral rules: (i) *cohesion*, which drives an agent toward to its neighbors; (ii) *repulsion*, which pushes an agent away from its neighbors to prevent collisions; and (iii) *alignment*, which aligns an agent to the dominant heading of its neighbors. For goal-oriented swarm motion, *alignment* can be replaced by a *migration* behavior so that each agent is directed in a desired orientation and moves at a preferred speed [44]. In complex environments, a fourth behavior called *collision avoidance* can be added to guide



(a) Pure formation control

(b) Proposed approach

Figure 3.1: We develop an event-based reconfiguration control (ERC) method to guide a TVF through narrow spaces. *Left:* Motion path of the TVF using purely behavior-based formation control [7], [12], which is collision with surrounding obstacles. *Right:* Motion path of the TVF using our proposed approach, which safely navigates through the narrow spaces.

the agent around obstacles [9], [27], [45], [46]. These behavioral rules are typically integrated into multi-robot systems through virtual forces within the artificial potential field (APF) to achieve coordinated movements [27], [47], [48]. Specifically, the APF has been used in [9], [46], [49] to navigate robot swarms through different environments, from open spaces to environments containing concave obstacles. In [9], a fuzzy controller is added to allow the robot swarm to avoid complex obstacles while maintaining swarm connectivity. In [12], evolutionary optimization is combined with a behavioral model to enable stable and decentralized navigation for large-scale aerial robot swarms in confined spaces. However, using a fixed set of behaviors limits the flexibility of the swarm formation in response to abrupt changes in the environment, especially in spaces such as caves, corridors, and tunnels, and hence increases the risk of collisions [50], [51].

Apart from the behavior-based approach, studies in formation control can be further grouped into two primary categories: *rigid formation* [50], [52]–[54] and *adaptive formation* [11], [31], [51], [55]. In *rigid formation*, the swarm can shrink or expand in size while preserving its overall shape. In [50], a model predictive control is employed for rigid formation in which optimal control signals are generated for each robot in the swarm based on a shared map. In [53], a region-based hierarchical control method is introduced for obstacle avoidance in confined spaces. The controller drives the robots to move cohesively within a virtual circular region that can contract to navigate around obstacles. A comprehensive control scheme is presented in [54] for omnidirectional mobile robots to transport a plate through unknown environments collaboratively. The scheme uses graph-based path planning for obstacle avoidance and distributed model predictive control for optimal motion. Although rigid formations work well in open-space environments, they become ineffective in narrow spaces where the width of the space and the size of the formation directly affect navigation success. This challenge becomes critical when contracting the formation increases the risk of collision among the robots, as illustrated in Figure 3.1a.

In *adaptive formation*, the swarm can transform into different configurations in adaptation to environmental conditions. In [55], a reconfiguration strategy is combined with behavioral control for swarm navigation in dynamic environments. The strategy uses an auction-based market approach to generate solutions for formation movement. In [31],

particle swarm optimization is used to generate trajectories for a swarm of unmanned aerial vehicles (UAVs) that can reconfigure its formation for obstacle avoidance and task accomplishment. However, this approach is centralized and requires considerable computational resources to maintain an optimal configuration in real time. A robust adaptive formation control algorithm is introduced in [56] for a group of UAVs. The controller can steer the vehicles to form and maintain different formation patterns for flexible navigation. Generally, the adaptive formation methods are effective in guiding robots through complex environments as they enable flexible reconfiguration in response to environmental changes. However, there remains a need for a distributed, partial communication solution that ensures safe, efficient formation navigation while adapting effectively to environmental change.

In this work, we propose an event-based reconfiguration controller (ERC) for safe and effective navigation of a decentralized TVF in narrow environments, as given in Figure 3.1. The robots are equipped with local sensors and communication modules to collect information about the environment and other robots in the formation. The main contributions of this study are threefold:

1. Define a set of individual behaviors that meet the reconfiguration control requirements, including goal-directed motion, formation maintenance, tailgating, and collision avoidance behaviors. Each behavior is designed for convenience in implementation via an artificial potential field.
2. Propose an event-based reconfiguration controller with two modes, “*formation*” and “*tailgating*” capable of adapting the formation shape in response to environmental changes. The stability of the proposed approach has been demonstrated via the Lyapunov theorem.
3. Extensive simulations and comparisons have been conducted to evaluate the robustness, scalability, and effectiveness of the proposed controller. Software-in-the-loop tests have also been conducted to verify its practical applicability. The source code of the proposed controller is publicly available for further research and practical implementation.

The remaining sections of this chapter are organized as follows. Section 3.2 presents the formation model and formulation. Section 3.3 introduces the proposed event-based reconfiguration control method. Simulation, comparison, and software-in-the-loop experimental results are shown in Section 3.4. The paper ends with conclusions drawn in Section 3.5.

## 3.2 Preliminaries

### 3.2.1 Robot Model

Consider a swarm  $\mathcal{N}$  that contains  $n$  robots labelled  $i \in \{1, \dots, n\}$ . The swarm is modeled as a directed sensing graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , where vertex set  $\mathcal{V} = \{1, \dots, n\}$  represents the robots, and edge set  $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$  contains robot pairs  $(i, j) \in \mathcal{E}$  in which robot  $i$  can

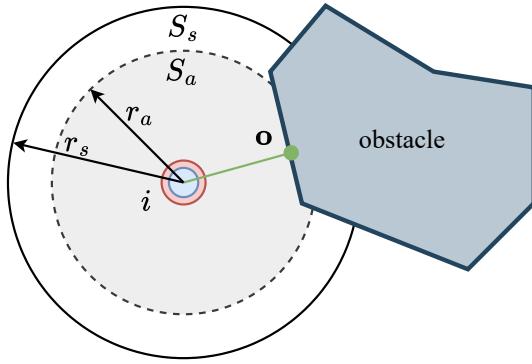


Figure 3.2: Illustration of a robot with a local range sensor. Each robot is equipped with a local sensor with sensing area  $S_s$  (solid white circle) being a circular disk within radius  $r_s$ . Additionally, alert area  $S_a$  (dashed gray circle) is a circular disk within radius  $r_a$ , with  $r_a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

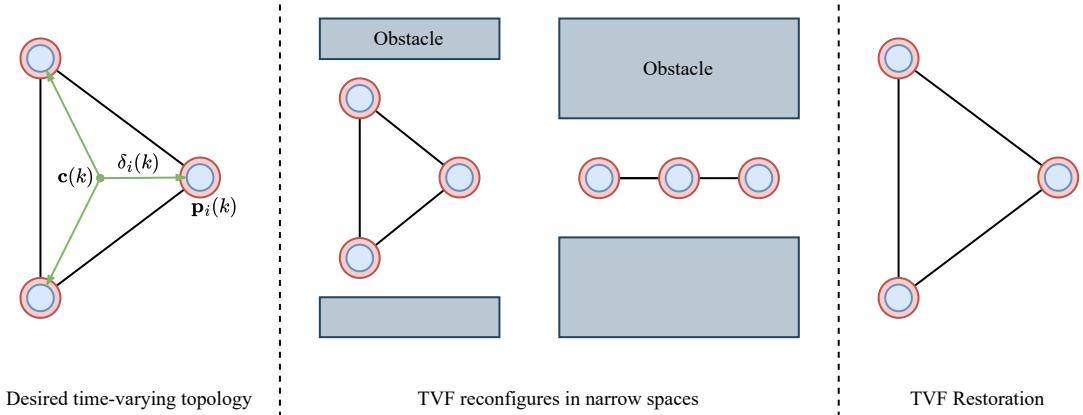


Figure 3.3: Schematic diagram of time-varying formation in the narrow spaces

sense robot  $j$ . Denote  $\mathcal{N}_i = \{j \in \mathcal{V} | (i, j) \in \mathcal{E}\} \subset \mathcal{V}$  as the set of  $n_i$  neighbors of robot  $i$  in  $\mathcal{G}$ .

In this work, robots in the swarm are identical, each with a body radius  $r$ . Each robot is equipped with an inertial measurement unit (IMU) to determine its position and orientation in a desired direction, a range sensor, and a wireless ad-hoc network module for peer-to-peer communication with other robots. The communication delay between any two robots is assumed to be negligible [11], [57]. The range sensor has  $360^\circ$  field of view providing a scanning area  $S_s$  of radius  $r_s$ , as depicted in Figure 3.2. The sensor data collected by robot  $i$  at time  $t(k)$  is represented as set  $\mathcal{M}_i(k)$  of data points  $\mathbf{o}$  to obstacles,  $\mathcal{M}_i(k) = \{\mathbf{o}\}$ . A safe area of radius  $r_a$  is defined so that repulsive forces are activated when robot  $i$  senses obstacles or its neighbors within range  $r_a$ .

The robot's dynamics is expressed in discrete time. Let  $\tau$  be the time step, and  $\mathbf{p}_i(k), \mathbf{v}_i(k), \mathbf{u}_i(k) \in \mathbb{R}^3$  be the position, velocity, and control input of robot  $i$  at time  $t(k) = k\tau$ , respectively. According to [40], the robot's dynamics can be expressed by a discrete linear system as follows:

$$\mathbf{x}_i(k+1) = \mathbf{A}_i \mathbf{x}_i(k) + \mathbf{B}_i \mathbf{u}_i(k), \quad (3.1)$$

where  $\mathbf{A}_i$  and  $\mathbf{B}_i$  are system matrices,  $\mathbf{u}_i$  is acceleration input, and  $\mathbf{x}_i = [\mathbf{p}_i, \mathbf{v}_i]^T \in \mathbb{R}^6$

is a state vector containing position and velocity. The velocities and accelerations are bounded so that  $\|\mathbf{v}_i(k)\| \leq v_{\max}$  and  $\|\mathbf{u}_i(k)\| \leq u_{\max}$ .

### 3.2.2 Problem formulation

This chapter addresses the time-varying formation (TVF) control for a swarm of robots in narrow spaces. Denote  $\delta(k)$  as the desired formation configuration at time  $k$ . The TVF problem can be described via the following definitions [40], [41].

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

$$\lim_{k \rightarrow \infty} \sum_{i=1}^n (\mathbf{p}_i(k) - \delta_i(k) - \mathbf{c}(k)) = 0 \quad (3.2)$$

where  $\mathbf{c}(k)$  is the formation center at time  $k$ .

**Definition 2.** *Safe formation: Given the TVF defined in Definition 1, this TVF is said to be safe for any robot  $i$ , with  $i \in \{1, \dots, n\}$  in the formation if the following conditions are satisfied:*

1. Formation configuration

$$\lim_{k \rightarrow \infty} \sum_{j=1, j \neq i}^n \|(\mathbf{p}_i(k) - \delta_i) - (\mathbf{p}_j(k) - \delta_j)\| = 0 \quad (3.3)$$

2. Safe distance between robots

$$\|\mathbf{p}_i(k) - \mathbf{p}_j(k)\| > 2r \quad (3.4)$$

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

3. Safe distance from obstacles

$$\|\mathbf{p}_i(k) - \mathbf{o}\| > r \quad (3.5)$$

for all  $\mathbf{o} \in \mathcal{M}_i(k)$ .

**Remark 1.** From Definition 1, the robots reach their desired positions, hence achieving the desired configuration. In the decentralized approach, when (3.3) is established, the system also ensures convergence to the desired configuration [40]. Conditions (3.4) – (3.5) ensure collision avoidance both among robots within the formation and between any robot and surrounding obstacles.

### 3.2.3 Formation configurations

Formation configurations refer to the shape that the robots form while cooperating and interacting with the environment. This work considers two following primary configurations:

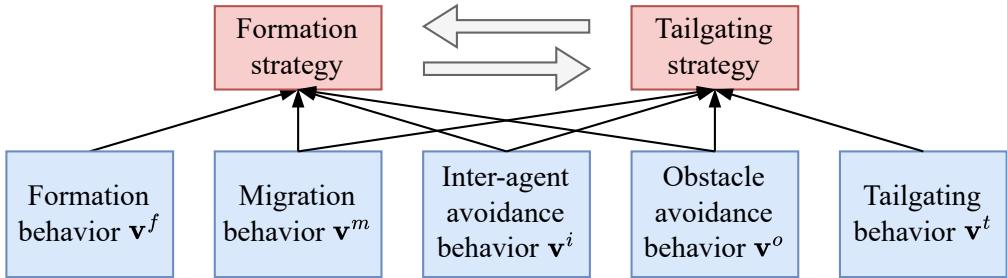


Figure 3.4: Overview of the proposed event-based reconfiguration control. The proposed strategy is constructed by two primary emergent strategy, including “*formation*” and “*tailgating*”, which are highlighted by red boxes. There are five individual behaviors that contribute to the emergent strategy,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

1. *Desired configuration*: The desired configuration represents the arrangement that the robot swarm is expected to maintain to accomplish its task. Common desired configurations include shapes such as polygons and V-shapes.
2. *Straight line configuration*: This formation configuration is employed when the robots do not have enough space to maintain their original configuration. It is constructed by having a robot follow and keep desired distance from its leader.

### 3.3 Event-based Reconfiguration Control

To address the TVF problem, 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 Figure 3.4.

Our approach incorporates an event-triggering mechanism into the TVF that allows the swarm to adapt its formation shape in response to environmental conditions for safe navigation, as illustrated in Figure 3.3. Initially, the desired configuration assigned to the TVF is denoted  $\delta_0$ . 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  $\delta_0$  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  $\delta_0$  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.

#### 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  $v_{\text{ref}}$  is the preferred speed and  $\mathbf{u}_{\text{ref}}$  is the preferred direction [44]. Then, the migration term, equal for every agent, corresponds to:

$$\mathbf{v}_i^m = v_{\text{ref}} \mathbf{u}_{\text{ref}} \quad (3.6)$$

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:

$$\mathbf{v}_i^f = k_f \sum_{j=1}^n (\mathbf{p}_j - \mathbf{p}_i - \kappa (\delta_j - \delta_i)) \quad (3.7)$$

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  $d_{\text{ref}}$ , with  $d_{\text{ref}} > 2r$ , an attractive force field for robot  $i$  can be expressed as follows:

$$\mathbf{v}_i^t = \begin{cases} k_t \left( \mathbf{p}_{l_i} - \mathbf{p}_i - d_{\text{ref}} \frac{\mathbf{v}_{l_i}}{\|\mathbf{v}_{l_i}\|} \right) + \mathbf{v}_{l_i} & \text{if } l_i \neq -1 \\ 0 & \text{otherwise} \end{cases} \quad (3.8)$$

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

$$\tilde{\mathbf{p}}_{ij} = \langle (\mathbf{p}_j - \mathbf{p}_i), \mathbf{u}_{\text{ref}} \rangle \quad (3.9)$$

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

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

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 Figure 3.2. Those repulsive forces are given as follows:

$$\mathbf{v}_i^i = k_i \sum_{j=1, j \neq i}^n \mathbf{v}_{ij}^i \quad (3.11)$$

$$\mathbf{v}_i^o = k_o \sum_{\mathbf{o} \in \mathcal{M}} \mathbf{v}_{io}^o \quad (3.12)$$

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

$$\mathbf{v}_{ij}^i = \begin{cases} \frac{r_a - \|\mathbf{p}_{ij}\|}{r_a - 2r} \hat{\mathbf{p}}_{ij} & \text{if } \|\mathbf{p}_{ij}\| < r_a \\ 0 & \text{otherwise} \end{cases} \quad (3.13)$$

$$\mathbf{v}_{io}^o = \begin{cases} \frac{r_a - \|\mathbf{p}_{io}\|}{r_a - r} \hat{\mathbf{p}}_{io} & \text{if } \|\mathbf{p}_{io}\| < r_a \\ 0 & \text{otherwise} \end{cases} \quad (3.14)$$

### 3.3.2 Event-based Reconfiguration Control strategy

As mentioned in Figure 3.4, the proposed event-based reconfiguration 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:

$$\tilde{\mathbf{v}}_i = \begin{cases} \mathbf{v}_i^f + \mathbf{v}_i^m + \mathbf{v}_i^c + \mathbf{v}_i^o & \text{if mode} = \text{"formation"} \\ \mathbf{v}_i^t + \mathbf{v}_i^m + \mathbf{v}_i^c + \mathbf{v}_i^o & \text{if mode} = \text{"tailgating"} \end{cases} \quad (3.15)$$

To search the large parameter space  $\Xi = \{k_f, k_t, k_i, k_o\}$  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  $\mathbf{u}_{\text{ref}}$  direction. Denote  $\mathbf{o}_l$  and  $\mathbf{o}_r$  are the nearest obstacle points on the left and right sides, respectively, whose distance to the robot  $i$  is minimum. If  $\mathbf{o}_l$  and  $\mathbf{o}_r$  exist, the width of the environment can be estimated by the magnitude of the cross product of the vector  $(\mathbf{o}_r - \mathbf{o}_l)$  and the desired direction  $\mathbf{u}_{\text{ref}}$  as follows:

$$w_e = \|(\mathbf{o}_r - \mathbf{o}_l) \times \mathbf{u}_{\text{ref}}\| \quad (3.16)$$

The estimation of the environment's width can be depicted in Figure 3.5. Besides, the formation's width  $w_f$  can be easily determined through the predefined formation

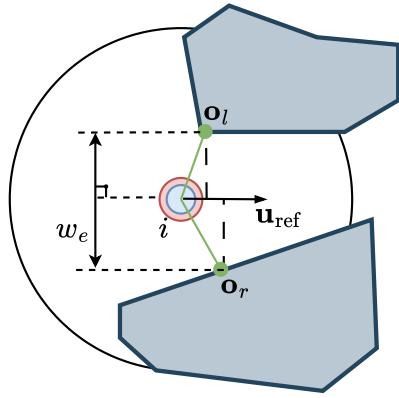


Figure 3.5: Environment's width estimation

topology. Given the widths of the environment and formation, the scaling factor  $\kappa$ , which contributes to (3.7), can be computed as follows:

$$\kappa = \begin{cases} \frac{w_e - 2r}{w_f} & \text{if } w_e \geq \lambda r \\ 0 & \text{otherwise} \end{cases} \quad (3.17)$$

As a result, each robot  $i$  can choose its mode and compute scaling factor  $\kappa$ . 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.

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

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

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

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

### 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  $\mathbf{L}$  be the Laplacian matrix of the interaction topology graph of the swarm. Due to the fully connected, the entries of  $\mathbf{L} = [l_{ij}]$  are given as follows:

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

---

**Algorithm 1:** Pseudocode of the ERC strategy

---

```

1 Get a set of observed obstacle points  $\mathcal{M}_i$ ;
2 if  $\mathcal{M}_i$  is empty then
3   mode  $\leftarrow$  "formation";
4    $\kappa \leftarrow 1.0$ ;
5 else
6   Determine the obstacle points  $\mathbf{o}_l$ ,  $\mathbf{o}_r$ ;
7   if  $\mathbf{o}_l$  or  $\mathbf{o}_r$  then
8     mode  $\leftarrow$  "formation";
9      $\kappa \leftarrow 1.0$ ;
10  else
11    Get the space's width  $w_e$ ;                                /* Eq. 4.21 */
12    if  $w_e \leq \lambda r$  then
13      mode  $\leftarrow$  "tailgating";
14    else
15      mode  $\leftarrow$  "formation";
16      Estimate the desired formation width  $w_f$ ;
17      if  $w_e - 2r \leq w_f$  then
18        Compute the scaling factor  $\kappa$ ;                            /* Eq. 3.17 */
19      else
20         $\kappa \leftarrow 1.0$ ;
21  Compute desired velocity  $\tilde{\mathbf{v}}_i$ ;                                /* Eq. 3.15 */
22 return  $\tilde{\mathbf{v}}_i$ ;

```

---

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

$$k_f \sum_{j=1}^n (\mathbf{p}_j - \mathbf{p}_i - \kappa \delta_{ji}) = k_f \sum_{j=1}^n (\mathbf{p}_j - \mathbf{p}_i) + \mathbf{b}_i \quad (3.21)$$

where the bias  $\mathbf{b}_i = -k_f \sum_{j=1}^n \kappa \delta_{ji}$ . 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:

$$\dot{\mathbf{P}} = -k_f \mathbf{LP} + \mathbf{B} \quad (3.22)$$

where  $\mathbf{P} = [\mathbf{p}_1, \dots, \mathbf{p}_n]^T$ ,  $\mathbf{B} = [\mathbf{b}_1, \dots, \mathbf{b}_n]^T$  and  $\mathbf{L}$  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  $\mathbf{B}$  is an eigenvector of  $\mathbf{L}$  with the corresponding eigenvalue  $n$  and:

$$\mathbf{LB} = n\mathbf{B} \quad (3.23)$$

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

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

Taking the first derivative of  $V_f$  gives

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

According to the LaSalle's invariance principle, it can be stated that as  $k \rightarrow \infty$  the state  $\mathbf{P}$  will converge to the largest invariant subset  $\Omega = \{\mathbf{P} | \dot{V}_f = 0\}$ . 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:

$$V_t = \frac{1}{2} \left( \mathbf{p}_{l_i} - \mathbf{p}_i - d_{\text{ref}} \frac{\mathbf{v}_{l_i}}{\|\mathbf{v}_{l_i}\|} \right)^T \left( \mathbf{p}_{l_i} - \mathbf{p}_i - d_{\text{ref}} \frac{\mathbf{v}_{l_i}}{\|\mathbf{v}_{l_i}\|} \right)\tag{3.26}$$

Taking the first derivative of  $V_t$ , gives:

$$\dot{V}_t = \left( \mathbf{p}_{l_i} - \mathbf{p}_i - d_{\text{ref}} \frac{\mathbf{v}_{l_i}}{\|\mathbf{v}_{l_i}\|} \right)^T (\mathbf{v}_{l_i} - \mathbf{v}_i)\tag{3.27}$$

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

$$\dot{V}_t = -k_t \left( \mathbf{p}_{l_i} - \mathbf{p}_i - d_{\text{ref}} \frac{\mathbf{v}_{l_i}}{\|\mathbf{v}_{l_i}\|} \right)^T \left( \mathbf{p}_{l_i} - \mathbf{p}_i - d_{\text{ref}} \frac{\mathbf{v}_{l_i}}{\|\mathbf{v}_{l_i}\|} \right) \leq 0\tag{3.28}$$

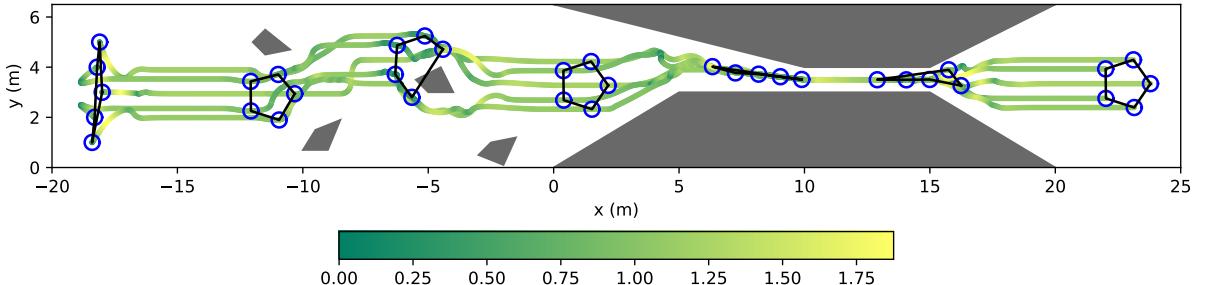
Therefore, the Lyapunov stability is satisfied. That leads to robot  $i$  converges to its leader  $l_i$  and keep behind a distance  $d_{\text{ref}}$  along its leader direction. As a result, the straight line configuration can be guaranteed.  $\square$

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

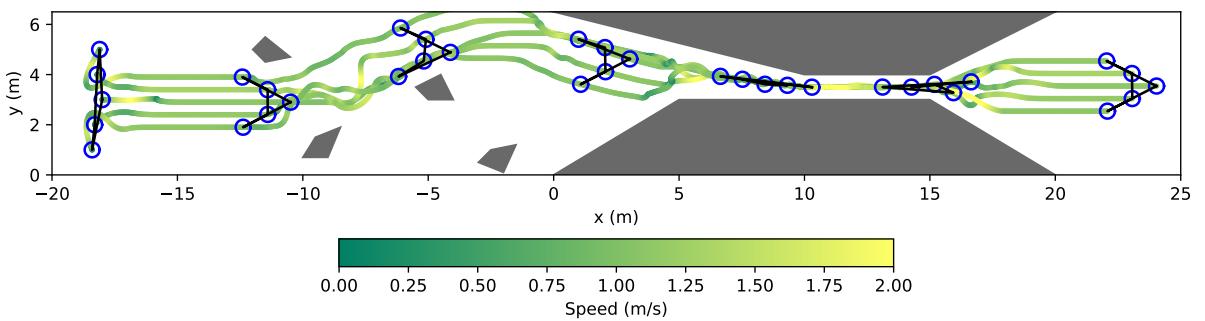
## 3.4 Results

### 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-like environment, whose obstacle densities is 0.05 obs/m<sup>2</sup> (for  $-12 \text{ m} < x < 0 \text{ m}$ ), another area is the width-varying cave-like environment with the most narrowed space is 1 m (for  $0 \text{ m} < x < 20 \text{ m}$ ). The TVF starts

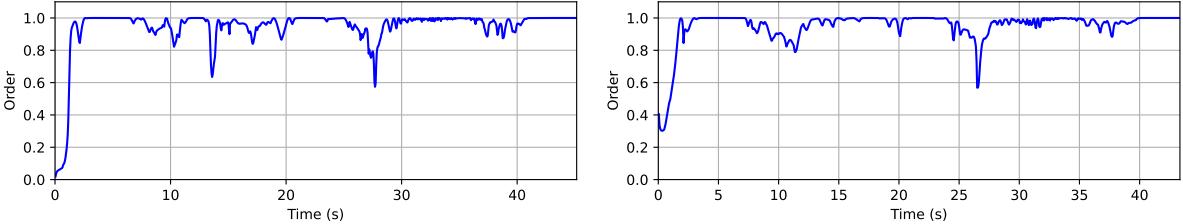


(a) Pentagon configuration



(b) V-shape configuration

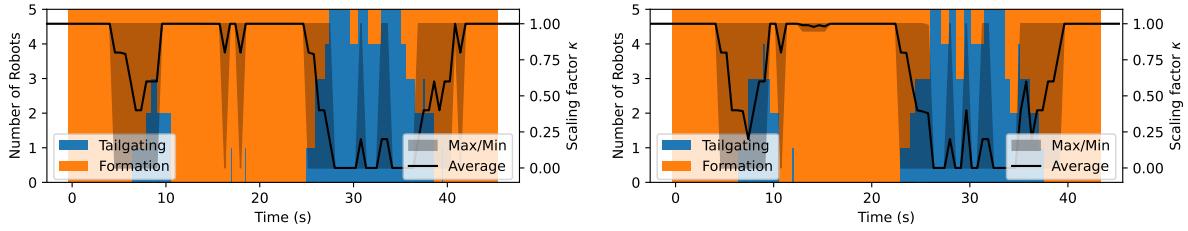
Figure 3.6: Motion paths and velocity profiles of the proposed ERC strategy in multiple configurations.



(a) Pentagon configuration

(b) V-shape configuration

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



(a) Pentagon configuration

(b) V-shape configuration

Figure 3.8: Correlation of robot number in each mode (bar chart) and scaling factor (black line) of the proposed ERC strategy

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  $\mathbf{u}_{\text{ref}} = [1, 0, 0]^T$ . We set up a formation with 5 homogeneous robots, each having the same radius  $r = 0.3$  m, the sensing radius  $r_s = 3.0$  m, and the alert radius  $r_a = 3r$ . The formation includes

two configurations, 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<sup>2</sup>. 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 ( $\sum \|\mathbf{u}(k)\|^2/nT$ ), 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 ERC guided a TVF to navigate through confined spaces. The motion paths and corresponding velocity profile are presented in Figure 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 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. Figure 4.4 presents 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*  $\Phi$  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].

$$\Phi = \frac{1}{n} \left\| \sum_{i=1}^n \frac{\mathbf{v}_i}{\|\mathbf{v}_i\|} \right\| \quad (3.29)$$

where  $v_i$  is the velocity of robot  $i$ .

Figure 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.

Figure 3.8 describes the correlation between the number of robots in different modes and the scaling factor  $\kappa$  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  $\kappa$  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

Table 3.1: Comparison between BC and our method, ERC. 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	Success	Mean speed (m/s)	Mean Acc. cost (m <sup>2</sup> /s <sup>3</sup> )
Pentagon	ERC	<b>10/10</b>	0.6773	<b>3.6873</b>
	BC	0/10	<b>0.7785</b>	4.3874
V-shape	ERC	<b>10/10</b>	0.7129	<b>4.0978</b>
	BC	0/10	<b>0.8877</b>	4.9402

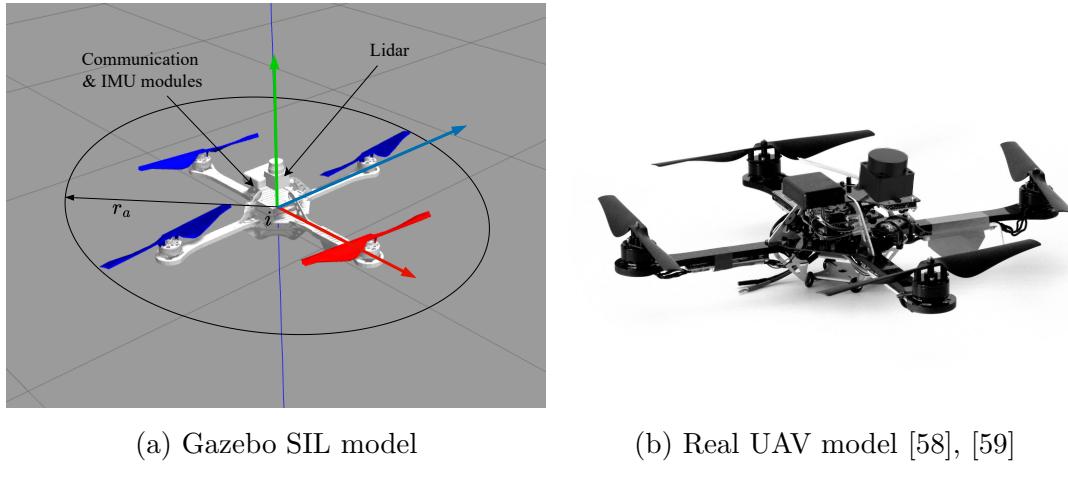


Figure 3.9: Used Hummingbird UAV model

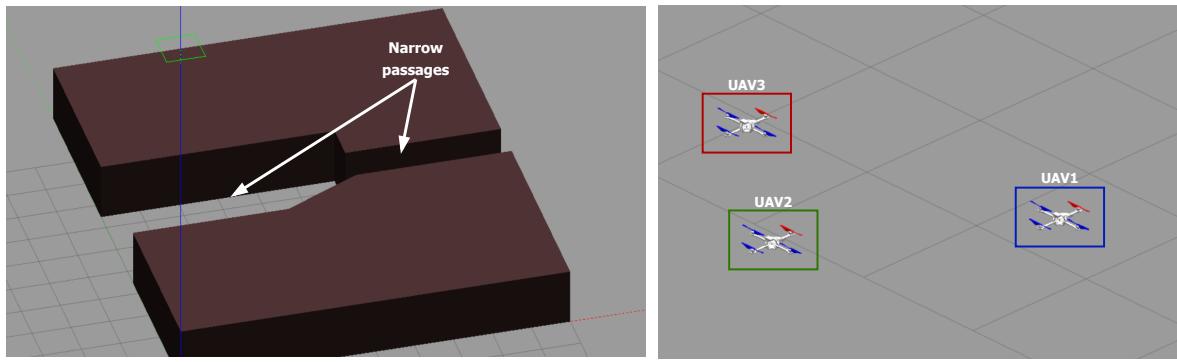
through narrow space. The value of  $\kappa = 0$  when all robots in the TVF are in the “tailgating” mode.

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, ERC and BC. It can be seen that the proposed ERC 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 Figure 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 ERC 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

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<sup>1</sup>. The used UAV model is a Hummingbird quadrotor developed based on Gazebo-based RotorS simulator [58], as described in Figure 3.9a. We set up three UAVs in the experiment with

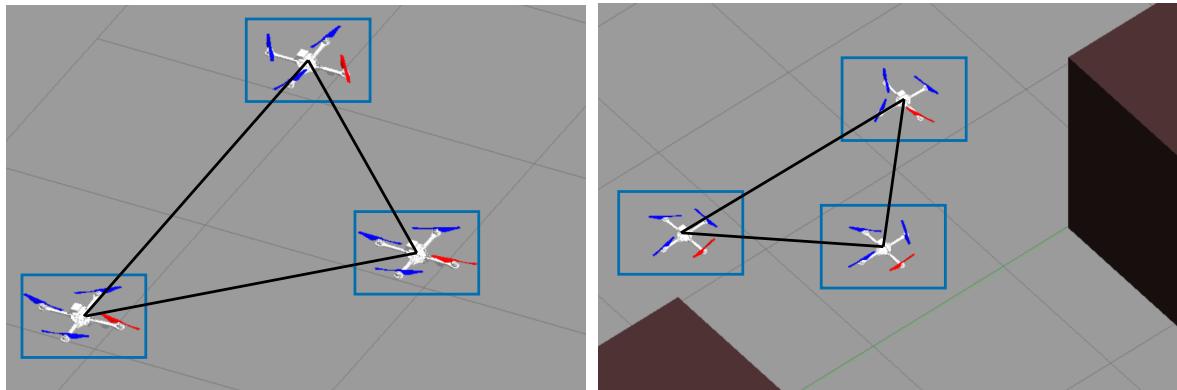
<sup>1</sup>Gazebo experiment video: <https://youtu.be/AIAAzRiIepg>



(a) The Gazebo environment

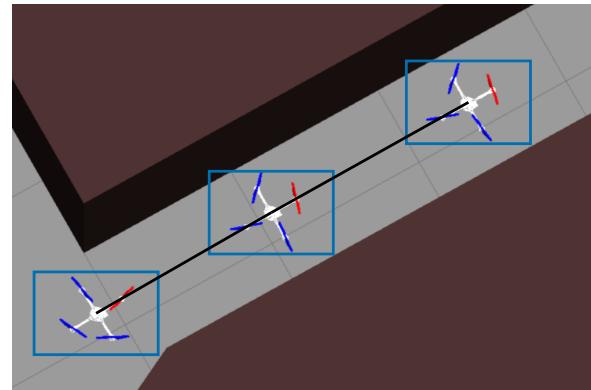
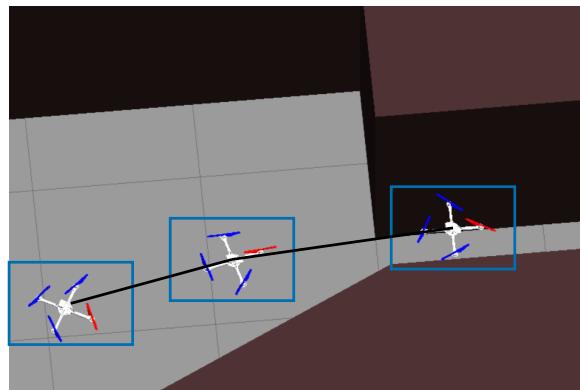
(b) Random initial positions

Figure 3.10: The environment in SIL test



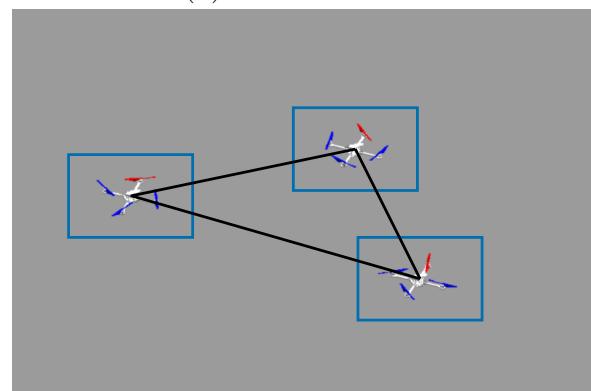
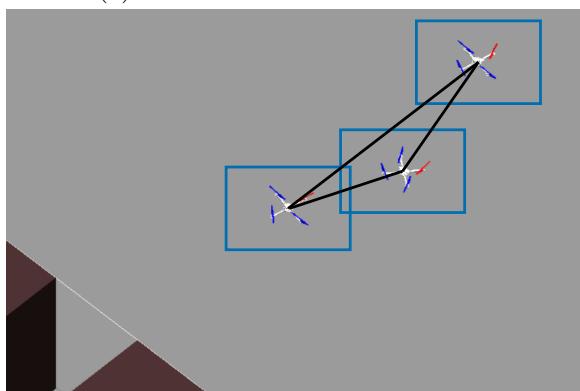
(a) Maintain triangle formation from random

(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

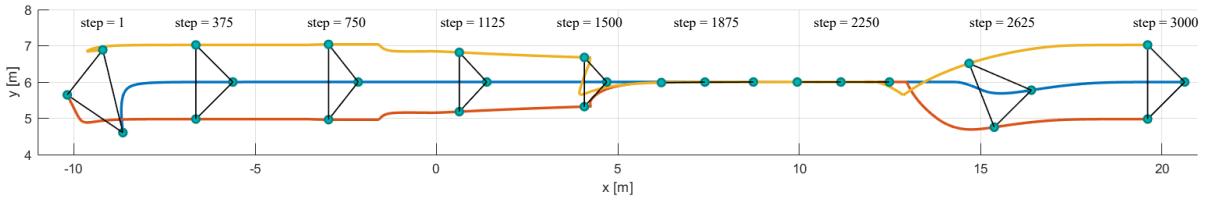


Figure 3.12: The recorded paths of three UAV in the SIL test (Top view)

random positions, as shown in Figure 3.10b. The environment in the SIL test includes two large obstacles, forming a width-varying tunnel, as depicted in Figure 3.10.

Figure 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 Figure 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 Figure 3.11a. When sensing the narrow passage, the formation shrinks to safely move through the passage (steps 1125 - 1500), as shown in Figure 3.11b. When the narrow space is not suitable for the original formation, the formation transforms to the straight line formation as Figure 3.11c and then passes through the space with the line formation (steps 1875 - 2550), as Figure 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 Figure 3.11e and moves to the target in Figure 3.11f (steps 2625 - 3000). The experiment demonstrates that the proposed ERC strategy successfully navigates the UAVs' formation to the target by passing through the narrow passage.

### 3.5 Conclusion

In this study, we have presented an event-based reconfiguration control to navigate a time-varying robot formation through narrow spaces across various environmental settings, including forest-like and tunnel-like terrains. The proposed strategy is constructed via several behaviors, which leverage the artificial potential field, which brings it to facilitate implementation. The controller includes two modes, “*formation*” and “*tailgating*”, and a scaling factor that enables the TVF to adapt its shape based on observed environmental conditions. The stability of the proposed approach is also demonstrated via the Lyapunov theorem. Evaluation results show that the proposed controller effectively navigates complex environments, achieving superior control performance in most metrics compared to other established methods. Additionally, software-in-the-loop tests verify the practicality and robustness of the proposed controller in real-world scenarios.

# Chapter 4

## Predictive Reconfiguration Control for Multi-Robot Formation in Cluttered Environments

*Manuscript*

Code: <https://github.com/duynamrcv/prc>

Video: <https://youtu.be/oAhg00hFSCA>

This chapter presents an optimal solution of the ERC, which is presented in Chapter 3 called predictive reconfiguration control (PRC) to navigate a swarm of robots through cluttered environments with narrow passages such as valleys or tunnels. A set of cost functions is introduced to represent swarm constraints, including formation shape, reference speed, movement direction, and collision avoidance. These cost functions also allow for predicting the swarm's states so that model predictive control solvers can be used to minimize the cost for optimal control signals.

### 4.1 Introduction

Formation control of multiple robots is becoming increasingly important due to its capability to perform challenging tasks in complex environments such as disaster response, search and rescue, tracking of multiple targets, and swarm-based reconnaissance [1], [4]. This approach also enhances system resilience as any malfunction within the swarm can be mitigated by reconfiguring the formation. The reconfiguration also enables the swarm to navigate through cluttered environments where a fixed topology is insufficient to adapt the formation to varying conditions. Formation reconfiguration control therefore is essential to ensure the effective operation of a robot swarm.

In formation control, a popular approach is based on natural collectives, such as school of fish or flocking of birds [60], where movement is driven by a set of simple behaviors: *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 *obstacle repulsion* to avoid collisions [42]. Those behaviors are often used with artificial potential fields (APF) to produce control signals for each robot [7]–[9], [61]. In particular, formation and navigational behaviors are combined in [7] as concurrent asynchronous processes to guide a group of robots to reach their goals in a desired formation. In [8], the APF is used to coordinate behaviors among fish-inspired miniature underwater robots to obtain

collective capabilities without having direct communication. However, behavior-based methods are limited in guiding the swarm through complex environments because a set of fixed behaviors lack the flexibility to handle abrupt changes in the environment [62].

To address this limitation, several studies propose a hybrid approach that combines the AFP with optimization techniques to better integrate behaviors. For example, the APF is used with an artificial neural network in [10] to optimize potential force parameters so that the swarm can adapt to environmental conditions. Similarly, an evolutionary optimization framework is introduced in [12] to select parameters and fitness functions that enhance the velocity and cohesion of the swarm. Nevertheless, optimizing parameters is insufficient to address the main problem of the behavior-based approach, which depends on a set of fixed behaviors.

Recently, an approach using optimal control to manage swarm constraints and predict agents' states has been introduced for reliable formation [63]–[65]. In [13], [14], optimization-based motion planners are used for individual point-to-point transitions to maintain the formation shape while avoiding collisions in multi-robot systems. In [64], model predictive control (MPC) is introduced for swarm navigation in which cost functions and constraints are used to regulate speed, ensure safety, and guide the drones through the environment. The system however is centralized and dependent on a primary agent. A distributed MPC in [66] offers decentralized computation for formation control, but it is only practical in spacious cluttered environments. Maintaining a specific formation shape in tight spaces is challenging due to conflicts between shape preservation and collision avoidance.

In another direction, several studies propose control techniques that reconfigure the formation in response to sudden changes in the environment structure [11], [46], [47], [67]. In [11], a distributed method capable of reconfiguring the formation is introduced for environments with both static and dynamic obstacles. The method computes a movable convex region from the robots' current positions and then repositions them within this region based on the desired movement direction. However, the transition relies on reassigning robots to predefined virtual points on the reference formation rather than utilizing information obtained from the environment. In [47], affine transformations are used to modify formation shapes for obstacle avoidance in dense environments, but they require information to be shared among all the robots. Our previous work [46] adapts a V-shape formation to guide a robot swarm through narrow corridors. The two wings of the V-shape can expand or contract to form a suitable shape based on real-time information about the available space. Nonetheless, this approach has not considered the physical limits of the robots, which could lead to infeasible control inputs and unexpected collisions in certain scenarios. Besides, the ability of each robot to make decisions based on local sensor data remains limited.

Building upon recent progress in multi-robot formation, this work introduces a novel technique named predictive reconfiguration control (PRC) for safe and effective navigation of a decentralized multi-robot swarm in cluttered environments. The robots are equipped with local sensors and communication modules to collect information about the environment and the states of their neighboring robots. The PRC is implemented on each robot to enable adaptive formation capabilities. The contributions of this work are threefold:

1. Model the formation as a directed sensing graph where each node represents a robot capable of sensing its surroundings and communicating with its neighbors.

This representation allows the system to be decentralized and the formation to be formulated as an optimization problem.

2. Define a set of cost functions that represent the formation constraints and performance. The functions ensure not only the desired formation shape but also the swarm's performance, including the desired velocity, direction, and obstacle avoidance. The cost functions are designed to be compatible with existing solvers for MPC, thereby simplifying the implementation.
3. Propose a predictive reconfiguration controller with two modes, “*formation*” and “*tailgating*”, capable of adapting the formation shape in response to environmental changes. Extensive simulations and comparisons have been conducted to evaluate the robustness, scalability, and effectiveness of the proposed controller. Software-in-the-loop tests have also been conducted to verify its practical applicability. The source code of the proposed controller is publicly available for further research and practical implementation.

The remaining sections of this chapter are organized as follows. Section 4.2 describes the formation model. Section 4.3 presents the proposed formation reconfiguration control method. Section 4.4 shows simulations, comparisons, and software-in-the-loop experimental results. The paper ends with conclusions drawn in Section 4.5.

## 4.2 Formation Background

Consider a swarm  $\mathcal{N}$  of  $n$  robots labeled  $i \in \{1, \dots, n\}$ . The swarm is modeled as a directed sensing graph  $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ , where vertex set  $\mathcal{V} = \{1, \dots, n\}$  represents the robots, and edge set  $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{V}$  includes robot pairs  $(i, j) \in \mathcal{E}$  for which robot  $i$  can sense robot  $j$ . Denote  $\mathcal{N}_i = \{j \in \mathcal{V} | (i, j) \in \mathcal{E}\} \subset \mathcal{V}$  as the set of  $n_i$  neighbors of a robot  $i$  in  $\mathcal{G}$ .

In this work, the dynamics of the robots are represented in discrete time. Denote  $\mathbf{p}_i(k), \mathbf{v}_i(k), \mathbf{u}_i(k) \in \mathbb{R}^3$  respectively be the position, velocity and control input of robot  $i$  at time  $t(k) = k\tau$ , where  $\tau$  is the sampling period. The robots in the swarm are homogeneous with a body radius  $r$ . Each robot is equipped with an inertial measurement unit (IMU) to determine its position and orientation, a range sensor to scan the environment, 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], [57]. The range sensor provides a 360° field of view with the scanning area  $S_s$  of radius  $r_s$ , as shown in Figure 4.1. Its point data obtained at time  $t(k)$  is represented by set  $\mathcal{M}_i(k) = \{\mathbf{m}\}$ .

According to [64], the robot in the swarm can be represented as a discrete linear system as follows:

$$\mathbf{x}_i(k+1) = \mathbf{A}_i \mathbf{x}_i(k) + \mathbf{B}_i \mathbf{u}_i(k), \quad (4.1)$$

where  $\mathbf{A}_i$  and  $\mathbf{B}_i$  are system matrices,  $\mathbf{u}_i$  is input acceleration, and  $\mathbf{x}_i = [\mathbf{p}_i; \mathbf{v}_i] \in \mathbb{R}^6$  is a state vector including position and velocity. The velocities and accelerations are bounded, i.e.,  $\mathbf{v}_{\min} \leq \mathbf{v}_i(k) \leq \mathbf{v}_{\max}$  and  $\mathbf{u}_{\min} \leq \mathbf{u}_i(k) \leq \mathbf{u}_{\max}$ .

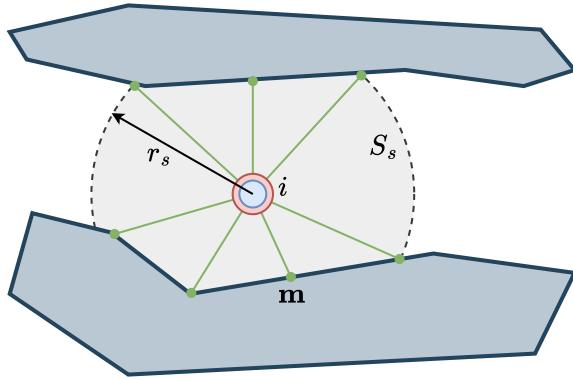


Figure 4.1: Illustration of a robot with its range sensor having the scanning area  $S_s$  (dashed gray circle) of radius  $r_s$  and set  $\mathcal{M}_i = \{m\}$  (green) of the acquired point data.

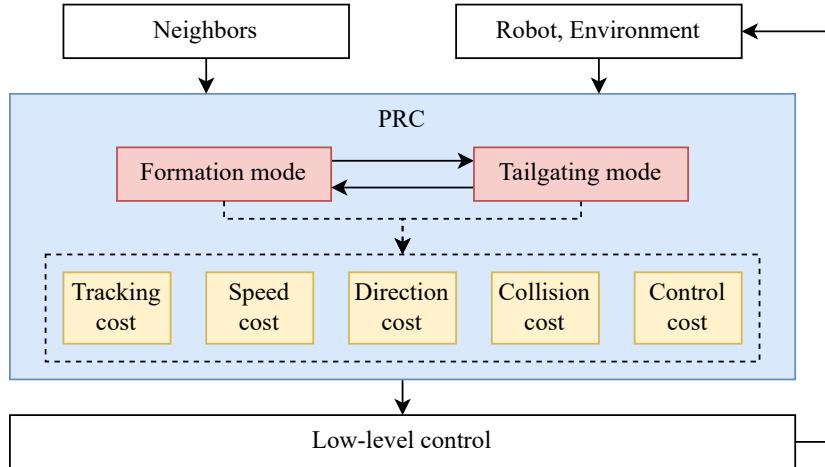


Figure 4.2: Diagram of the proposed predictive reconfiguration control strategy.

### 4.3 Predictive Reconfiguration Control

The aim of reconfiguration control is to drive the robot swarm through a cluttered environment having narrow passages. The swarm adheres to the following constraints: (C1) maintain certain desired shapes; (C2) move along a prioritized direction  $\mathbf{u}_{\text{ref}} \in \mathbb{R}^3$ ; (C3) obtain the desired speed  $v_{\text{ref}} \in \mathbb{R}$ ; and (C4) ensure no collision with other neighbors or obstacles in the environment. To address this problem, we propose a predictive reconfiguration control system as shown in Figure 4.2. Inputs to this system include point cloud data from range sensors and the states of neighbor robots. Depending on the environment structure inferred from the point cloud data, the controller operates in either the “formation” or “tailgating” mode. The “formation” mode maintains the desired shape, whereas the “tailgating” mode transforms the formation into a line to navigate through narrow spaces like valleys or tunnels. The controller is designed based on a weighted-sum of five cost functions to meet constraints (C1) – (C4). An optimal solver is then used to generate control signals for low-level controllers.

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  $j$ . The formation is obtained via the following constraint [41], [68]:

$$\lim_{k \rightarrow \infty} (\mathbf{p}_j(k) - \mathbf{p}_i(k) + \kappa \delta_{ij}) = 0, \quad \forall i, j \in \{1, \dots, n\}, i \neq j \quad (4.2)$$

where  $\kappa \in [0, 1]$  is a scaling factor representing the shrinkage level of the formation. The desired relative position of robot  $i$  in the formation is then described as:

$$\mathbf{p}_i^*(k) = \frac{1}{n_i} \sum_{j \in \mathcal{N}_i} (\mathbf{p}_j(k) + \kappa \delta_{ij}). \quad (4.3)$$

During the transition between modes, each robot determines a leader as a reference to determine its position. The leader is selected based on the inner product  $\tilde{\mathbf{p}}_{ij}$  of the difference between robot  $j$  in the neighbor set  $\mathcal{N}_i$  and robot  $i$ ,  $\mathbf{p}_j - \mathbf{p}_i$ , and the desired direction,  $\mathbf{u}_{\text{ref}}$ , as follows:

$$\tilde{\mathbf{p}}_{ij} = \langle (\mathbf{p}_j - \mathbf{p}_i), \mathbf{u}_{\text{ref}} \rangle. \quad (4.4)$$

A positive value of  $\tilde{\mathbf{p}}_{ij}$  indicates that robot  $j$  is in front of robot  $i$  in the  $\mathbf{u}_{\text{ref}}$  direction, and vice versa. Let  $\mathcal{P}_i$  be the set of inner products for all robots  $j$  in the neighbor set  $\mathcal{N}_i$ ,  $\mathcal{P}_i = \{\tilde{\mathbf{p}}_{ij}\}$ . Leader robot  $l_i$  of robot  $i$  is chosen as the closest robot in front of it, i.e.,

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

---

**Algorithm 2:** Pseudocode of the leader selection

---

```

1 foreach  $j \in \mathcal{N}_i$  do
2   Compute inner product  $\tilde{\mathbf{p}}_{ij}$ ;                                /* Eq. 4.4 */
3    $\mathcal{P}_i \leftarrow \tilde{\mathbf{p}}_{ij}$ ;
4 Select leader  $l_i$  for robot  $i$  to follow;                         /* Eq. 4.5 */
5 return  $l_i$ ;

```

---

Algorithm 2 presents the leader selection process.

In the “tailgating” mode, robot  $i$  needs to align and keep a distance  $d_{\text{ref}} \in \mathbb{R}$  with its leader  $l_i$ . This can be formulated as follows:

$$\lim_{k \rightarrow \infty} \|\mathbf{p}_{l_i}(k) - \mathbf{p}_i(k)\| = d_{\text{ref}} \quad (4.6)$$

The desired relative position of robot  $i$  then can be determined as follows:

$$\mathbf{p}_i^*(k) = \mathbf{p}_{l_i}(k) - d_{\text{ref}} \mathbf{u}_{\text{ref}}. \quad (4.7)$$

Using equations (4.4) and (4.7), the formation problem is converted into tracking the desired position  $\mathbf{p}_i^*$ , which can be handled by predictive controllers.

### 4.3.1 Predictive control design

The proposed predictive controller uses the desired position  $\mathbf{p}_i^*$  as the reference and a set of cost functions to fulfill formation constraints  $(C1) - (C4)$ . The functions include tracking cost  $J_{t,i}$ , direction cost  $J_{d,i}$ , speed cost  $J_{s,i}$ , obstacle avoidance cost  $J_{o,i}$ , inter-agent collision cost  $J_{i,i}$ , and control effort cost  $J_{u,i}$ . Let  $P \in \mathbb{N}^+$  be the prediction horizon, which is finite and shifts forward at each time step;  $(\cdot)(k+l|k)$ ,  $l \in \{0, \dots, P\}$ , be the

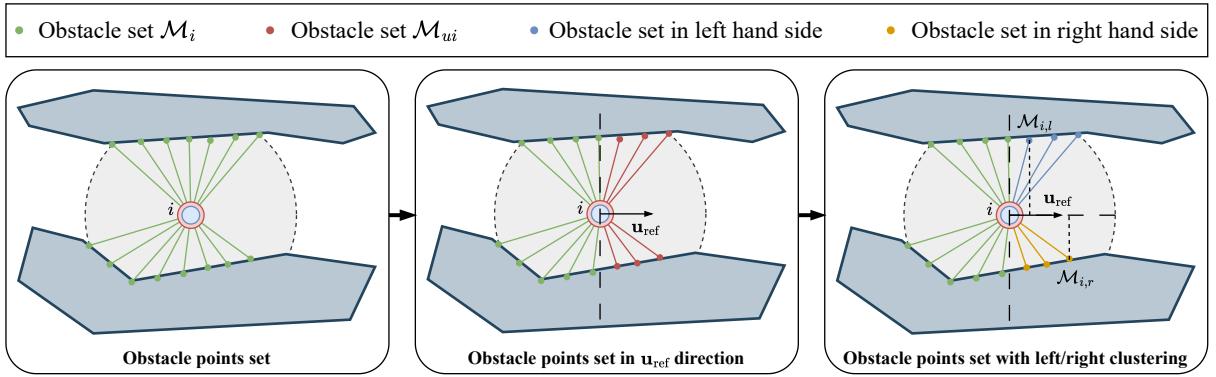


Figure 4.3: The process of estimating the environment's width from the robot's range sensor.

predicted value of  $(\cdot)(k+l)$  when the information at time  $t(k)$  is available;  $\mathbf{X}_i(k) \in \mathbb{R}^{6P}$  be the sequence of the predicted states  $\mathbf{x}_i(k+l|k)$  over the horizon  $l \in \{1, \dots, P\}$ ; and  $\mathbf{U}_i(k) \in \mathbb{R}^{3P}$  be the sequence of the predicted control inputs  $\mathbf{u}_i(k)$  over the horizon  $l \in \{0, \dots, P-1\}$ . The predictive reconfiguration control can be modeled as a non-convex optimization problem as follows:

$$\min_{\mathbf{U}_i(k)} (J_{t,i}(k) + J_{s,i}(k) + J_{d,i}(k) + J_{o,i}(k) + J_{i,i}(k) + J_{u,i}(k)) \quad (4.8)$$

subject to:

$$\begin{aligned} \mathbf{x}_i(k+l+1|k) &= \mathbf{A}\mathbf{x}_i(k+l|k) + \mathbf{B}\mathbf{u}_i(k+l|k), \\ \mathbf{x}_i(k|k) &= \mathbf{x}_i(k), \\ \mathbf{v}_{\min} &\leq \mathbf{v}_i(k+l|k) \leq \mathbf{v}_{\max}, \\ \mathbf{u}_{\min} &\leq \mathbf{u}_i(k+l|k) \leq \mathbf{u}_{\max}, \end{aligned} \quad (4.9)$$

with  $l \in \{1, \dots, P\}$ , and  $i \in \mathcal{N}$ . The cost functions are defined as follows.

### Tracking cost

The tracking term aims to drive the robots toward their reference positions to achieve the desired formation shape. It is defined as the square error between the desired position  $\mathbf{p}_i^*$  and the predicted position  $\mathbf{p}_i$  of robot  $i$  as follows:

$$J_{t,i}(k) = w_t \sum_{l=1}^P \|\mathbf{p}_i^*(k+l|k) - \mathbf{p}_i(k+l|k)\|^2, \quad (4.10)$$

where  $w_t$  is a positive tracking weight.

### Speed cost

The speed cost is used to maintain the desired speed  $v_{\text{ref}}$  of the swarm. It is defined as the squared difference between the actual and the desired speed of the robots as follows:

$$J_{s,i}(k) = w_s \sum_{l=1}^P \left( \|\mathbf{v}_i(k+l|k)\|^2 - v_{\text{ref}}^2 \right)^2, \quad (4.11)$$

where  $w_s$  is a positive weight.

## Direction cost

This cost directs the robots to move in the desired direction  $\mathbf{u}_{\text{ref}}$ . It is computed based on the normalized dot product between velocity  $\mathbf{v}_i$  and the desired direction  $\mathbf{u}_{\text{ref}}$  of robot  $i$ . It is equal to zero when the velocity perfectly aligns with the reference direction and otherwise increases proportionally with the degree of misalignment. It is given as follows:

$$J_{d,i}(k) = w_d \sum_{l=1}^P \left( 1 - \frac{\langle \mathbf{v}_i(k+l|k), \mathbf{u}_{\text{ref}} \rangle^2}{\|\mathbf{v}_i(k+l|k)\|^2} \right)^2, \quad (4.12)$$

where  $w_d$  is a positive direction weight.

## Collision avoidance cost

To avoid collisions, the distance from a robot to any obstacle must be greater than the robot's radius  $r$  and the distance between any two robots must be greater than  $2r$ . Let  $d_{ij} = \|\mathbf{p}_j - \mathbf{p}_i\|$  be the distance between robots  $i$  and  $j$ , and  $d_{im}$  be the distance between robot  $i$  and obstacle  $\mathbf{m}$ . The constraints for collision avoidance are given as follows:

$$d_{im}(k+l|k) \geq r \quad i \in \mathcal{N}, \mathbf{m} \in \mathcal{M}_i(k) \quad (4.13)$$

$$d_{ij}(k+l|k) \geq 2r \quad i \in \mathcal{N}, j \in \mathcal{N}_i \quad (4.14)$$

In this work, constraint (4.13) is represented via obstacle avoidance cost  $J_{o,i}$  defined as a logistic function as follows [69]:

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

where  $w_o > 0$  is a constant weight,  $\alpha > 0$  is a smoothness parameter, and

$$d_{im}^{\min}(k+l|k) = \min \{ d_{im}(k+l|k) | \mathbf{m} \in \mathcal{M}_i \}. \quad (4.16)$$

Similarly, constraint (4.14) is represented via an inter-agent collision cost  $J_{i,i}$  defined as follows [7]:

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

where  $w_i > 0$  is a constant weight and

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

with  $\beta > 2$  being the influence ratio of the neighbors.

## Control effort cost

The control effort cost is used as a penalty term to minimize the control signal. It is defined as:

$$J_{u,i}(k) = w_u \sum_{l=0}^{P-1} \|\mathbf{u}_i(k+l|k)\|^2, \quad (4.19)$$

where  $w_u > 0$  is a constant control weight.

### 4.3.2 Formation reconfiguration

Depending on the environment's width, the control system can determine the formation shape and scaling factor  $\kappa$ . The process of estimating the environment's width is illustrated in Figure 4.3. Robot  $i$  first obtains point cloud data  $\mathcal{M}_i$  (green) from its local sensor and then selects a point set  $\mathcal{M}_{ui}$  (red) in front of the robot along the moving direction  $u_{\text{ref}}$  as follows:

$$\mathcal{M}_{ui} = \{\mathcal{M}_i \mid \langle (\mathbf{p}_i - \mathcal{M}_i), u_{\text{ref}} \rangle < 0\}. \quad (4.20)$$

The DBSCAN algorithm [70] is then used to divide  $\mathcal{M}_{ui}$  into two clusters corresponding to the left (blue) and right (yellow) sides of the robot. Data points  $\mathcal{M}_{i,l}$  and  $\mathcal{M}_{i,r}$  from those clusters with the shortest distance to  $u_{\text{ref}}$  are then selected. Using these points, the environment's width is computed as:

$$w_e = \|(\mathcal{M}_{i,r} - \mathcal{M}_{i,l}) \times \mathbf{u}_{\text{ref}}\| \quad (4.21)$$

The pseudocode to estimate the environment's width is presented in Algorithm 3.

On the other hand, the formation's width  $w_f$  is predefined for each specific formation shape. The scaling factor  $\kappa$  then can be computed based on the environment's and formation's width as follows:

$$\kappa = \begin{cases} \frac{w_e - 2r}{w_f} & \text{if } w_e \geq \lambda r \\ 0 & \text{otherwise} \end{cases} \quad (4.22)$$

where  $\lambda > 2$  is a scaling coefficient determining the environment's width at which the PRC switches its mode.

---

#### Algorithm 3: Pseudocode to estimate the environment's width

---

- 1 Get point set  $\mathcal{M}_{ui}$  in front of robot  $i$  in moving direction  $\mathbf{u}_{\text{ref}}$ ; /\* Eq. 4.20 \*/
  - 2 Cluster  $\mathcal{M}_{ui}$  for the left and right sides of the robot using DBSCAN;
  - 3 Find point pair  $(\mathcal{M}_{i,l}, \mathcal{M}_{i,r})$ , whose distance to  $\mathbf{u}_{\text{ref}}$  is minimum;
  - 4 Compute the environment's width  $w_e$ ; /\* Eq. 4.21 \*/
  - 5 **return**  $w_e$ ;
- 

Algorithm 4 presents the pseudocode of the PRC. At each time step, the system estimates the width of the environment to determine the formation mode and the scaling factor  $\kappa$ . It then computes the desired position  $\mathbf{p}_i^*$  for each robot and establishes the formation cost function using equations (4.8) - (4.9). This problem is then solved to find the optimal control signal  $\mathbf{u}_i^*$  using common nonlinear programming (NLP) solvers, such as the Sequential Least Squares Programming (SLSQP) [72]. In this work, we implemented the solver in Python using the optimization software library SciPy [71].

---

**Algorithm 4:** Pseudocode of the PRC

---

```
1 Get data point set  $\mathcal{M}_i$ ;
2 if  $\mathcal{M}_i$  is empty then
3   mode  $\leftarrow$  “formation”;
4    $\kappa \leftarrow 1.0$ ;
5 else
6   Get the environment’s width  $w_e$ ;                                /* Alg. 3 */
7   if  $w_e$  is None then
8     mode  $\leftarrow$  “formation”;
9      $\kappa \leftarrow 1.0$ ;
10  else
11    if  $w_e \leq \lambda r$  then
12      mode  $\leftarrow$  “tailgating”;
13    else
14      mode  $\leftarrow$  “formation”;
15      Estimate the desired formation width  $w_f$ ;
16      if  $w_e - 2r \leq w_f$  then
17        Compute the scaling factor  $\kappa$ ;                                /* Eq. 4.22 */
18      else
19         $\kappa \leftarrow 1.0$ ;
20 switch mode do
21   case “formation” do
22     Get the desired position  $\mathbf{p}_i^*$ ;                                /* Eq. 4.3 */
23   case “tailgating” do
24     Select a leader to follow;                                         /* Alg. 2 */
25     Get the desired position  $\mathbf{p}_i^*$ ;                                /* Eq. 4.7 */
26 Establish the formation cost function;                                /* Eqs. 4.8-4.9 */
27 Minimize the cost function to obtain the optimal control signal  $\mathbf{u}_i^*$ ; /* MPC
   solver [71] */
28 return  $\mathbf{u}_i^*$ ;
```

---

## 4.4 Results and Discussion

A number of simulations, comparisons, and software-in-the-loop tests have been conducted to evaluate the performance of the PRC with details as follows.

### 4.4.1 Evaluation Setup

The swarm includes five identical robots, each with a radius of  $r = 0.2$  m, a maximum speed of  $\|\mathbf{v}_{\max}\| = 1.5$  m/s, and a maximum control input  $\|\mathbf{u}_{\max}\| = 2.0$  m/s<sup>2</sup>. The robot is equipped with a range sensor having the sensing range of  $r_s = 3$  m. The desired formation shape is set to a pentagon, the reference velocity is  $v_{\text{ref}} = 1$  m/s, and the desired

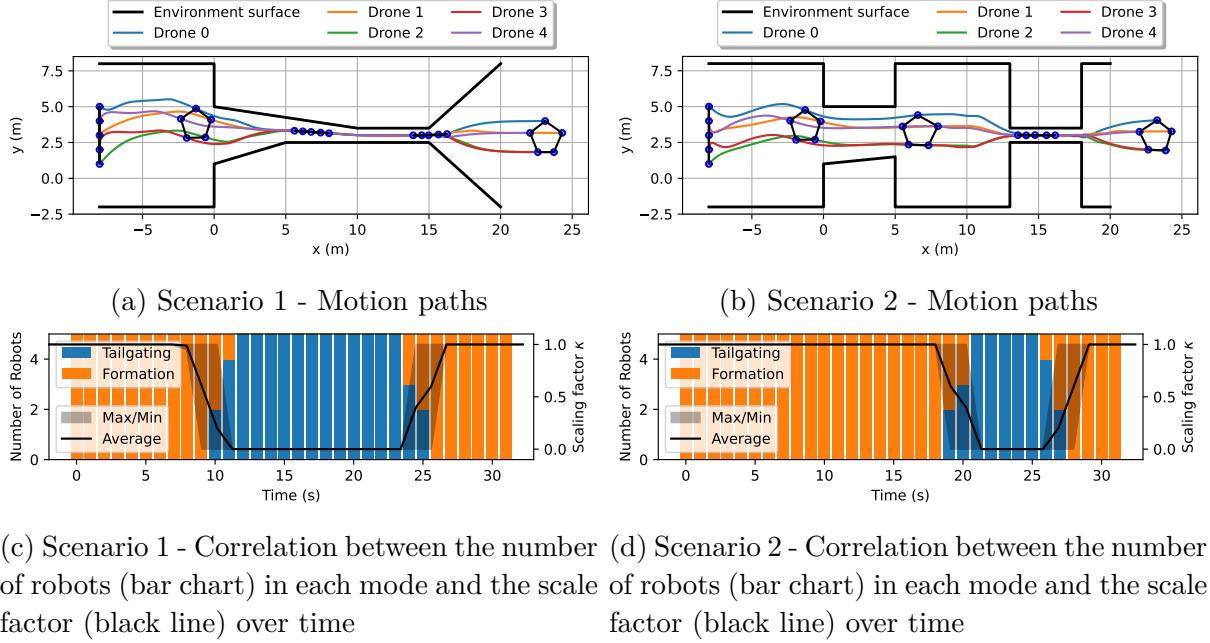


Figure 4.4: Trajectories and formation shapes of the robots controlled by the PRC in two evaluating scenarios.

direction is  $\mathbf{u}_{\text{ref}} = [1, 0, 0]^T$ . The environments include two structures, both having narrow passages, as depicted in Figure 4.4.

The metrics used for evaluation include the success rate, mean *order*  $\Phi$ , mean speed (m/s), mean formation error  $\varepsilon$  (m), and acceleration cost  $\Gamma$  ( $\text{m}^2/\text{s}^4$ ) [73]. The *order* metric [37] measures the heading consensus of the robots and is computed as:

$$\Phi = \frac{1}{N} \left\| \sum_{i=1}^N \frac{\mathbf{v}_i}{\|\mathbf{v}_i\|} \right\| \quad (4.23)$$

Its value ranges from 0 to 1, with 1 indicating the robots have the same direction. The *formation error* measures the deviation between the desired and actual positions of the robots and is calculated as [68]:

$$\varepsilon_i = \|\mathbf{p}_i - \mathbf{p}_i^*\| \quad (4.24)$$

The acceleration cost indicates the control effort and is given by:

$$\Gamma = \frac{1}{n} \sum_{i=1}^N \|\mathbf{u}_i(k)\|^2 \quad (4.25)$$

The comparing methods include the behavior-based reconfiguration control (BRC) [12] and the predictive formation control (PFC) [66]. In evaluation, each method is run 10 times for each scenario.

#### 4.4.2 Results

Figure 4.4 shows the formation results for both scenarios. After takeoff, the robots rapidly form the desired pentagon shape, then change their formation to pass through tight passages and finally transform back to the original desired shape. Figures 4.4c-d depict the scaling factor  $\kappa$  and the number of robots within each control mode

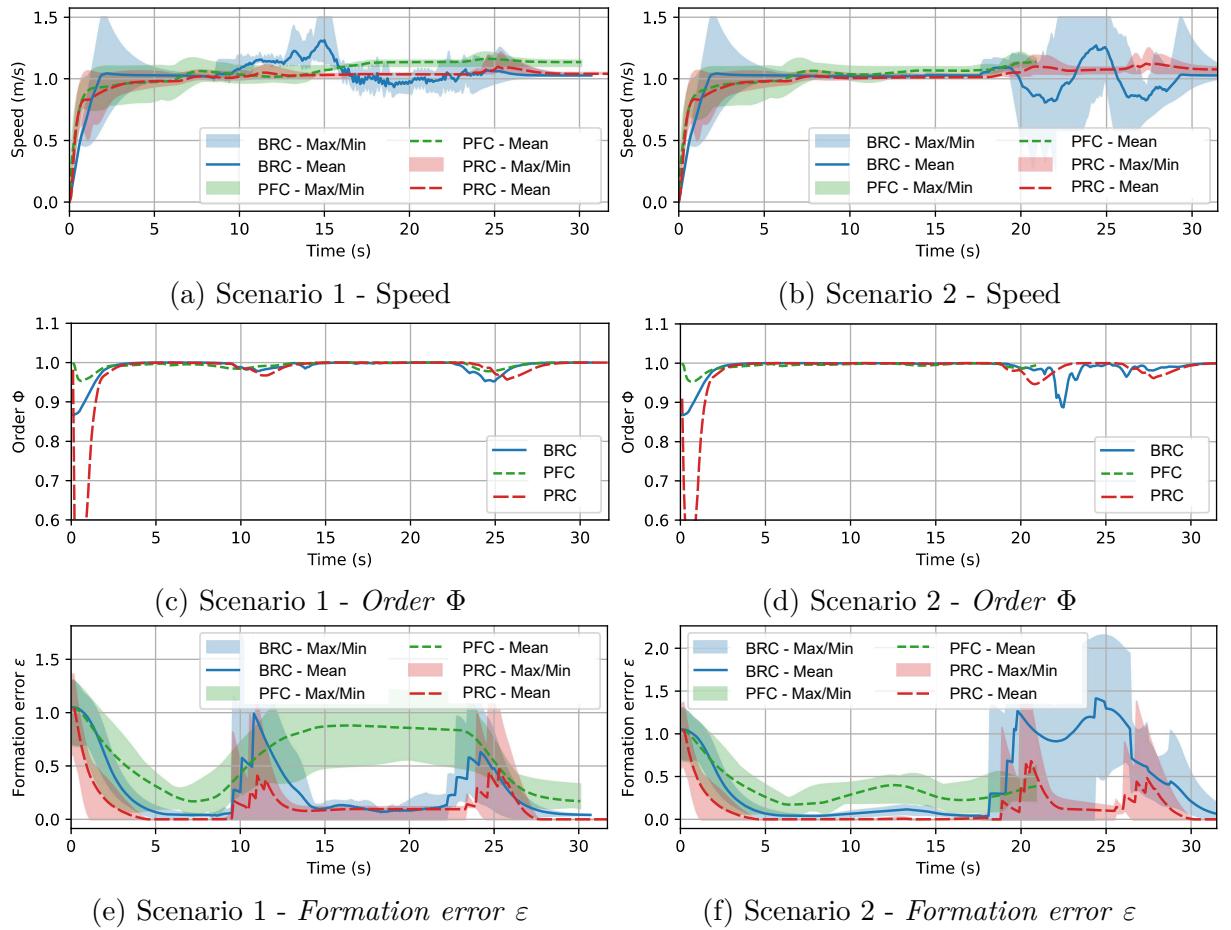


Figure 4.5: Comparison results of three control methods in two scenarios.

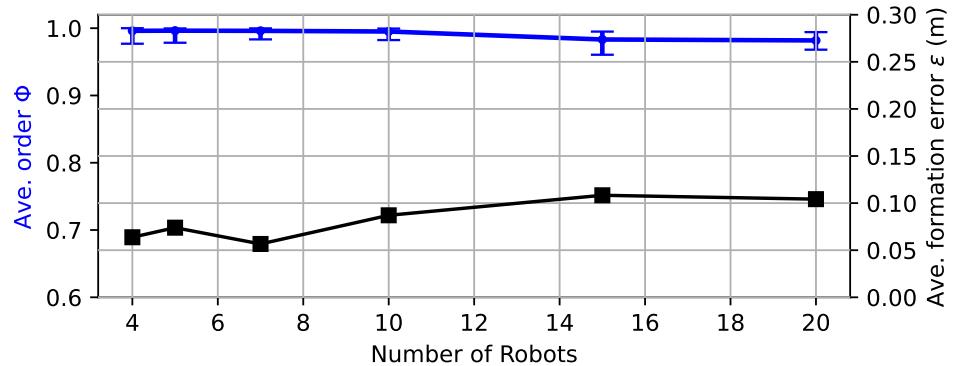


Figure 4.6: Effect of the swarm size on system performance, including the mean  $order \Phi$  and formation error  $\varepsilon$ .

Table 4.1: Comparison between BRC, PFC, and the proposed PRC

Scen.	Method	Success rate	Mean <i>order</i> $\Phi$	Mean speed (m/s) ( $v_{\text{ref}} = 1$ m/s)	Mean formation error $\varepsilon$ (m)	Acceleration cost $\Gamma$ (m <sup>2</sup> /s <sup>4</sup> )
1	BRC	<b>10/10</b>	0.9890	1.0249	0.3048	69.6589
	PFC	8/10	<b>0.9934</b>	1.0639	0.6376	<b>23.7442</b>
	PRC	<b>10/10</b>	0.9824	<b>0.9863</b>	<b>0.2423</b>	25.0894
2	BRC	6/10	0.9883	<b>0.9887</b>	0.6872	53.8718
	PFC	0/10	<b>0.9953</b>	1.0470	0.4593	<b>19.0365</b>
	PRC	<b>9/10</b>	0.9830	0.9800	<b>0.3217</b>	21.8559

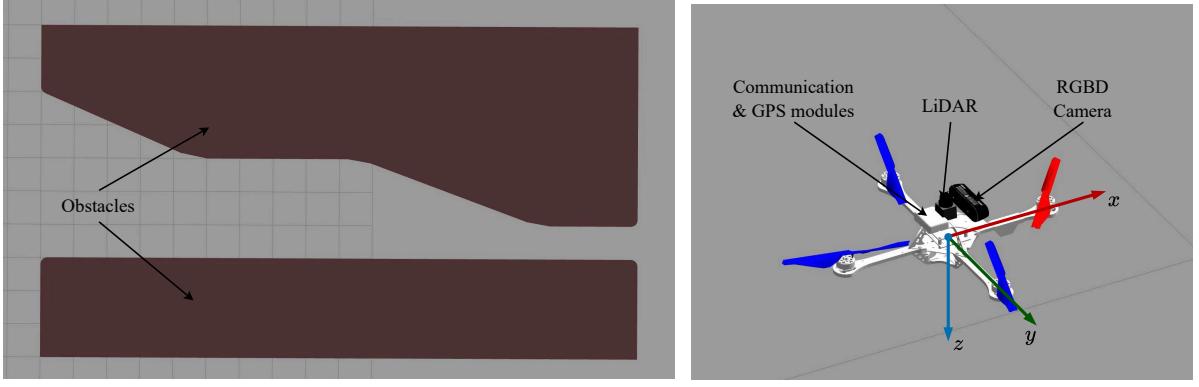
during operation. When the environment’s width is large, the scaling factor  $\kappa$  stays at 1 to maintain the desired shape. As the environment narrows,  $\kappa$  gradually decreases leading to more robots switching to “*tailgating*” mode. The formation thus shrinks until  $\kappa$  reaches 0, the point at which all robots are in “*tailgating*” mode. Upon exiting the narrow corridor,  $\kappa$  increases, allowing the formation to expand back to its original shape. The PRC hence provides reconfiguration capabilities for the robot swarm to adapt to complex environmental conditions.

Figure 4.5 presents comparison results between the PRC and other control methods. In terms of speed, the proposed controller achieves more stable velocities with values closer to the reference  $v_{\text{ref}}$  in both scenarios, as indicated via the average and maximum/minimum values shown in Figures 4.5a-4.5b. For the *order* metric, the PRC exhibits large fluctuation at start but quickly converges to a high consensus among the robots with the *order* value reaching 1, as shown in Figures 4.5c-4.5d. Both the BRC and PFC also perform well, although the BRC shows more variation than the PRC during the transition phase. In terms of formation maintenance, the PRC outperforms other methods with the smallest average error, as shown in Figures 4.5e-4.5f. These results are further confirmed in Table 4.1, which presents comparison data. The proposed PRC shows high performance in all metrics, with the highest success rate and the smallest formation error in both scenarios. The PFC fails in scenario 2 due to its limitation in reconfiguring the formation. The BRC has good performance in speed and mean order. It however introduces high acceleration costs, indicating that the method is not energy efficient.

In another experiment, we vary the swarm size between 4, 5, 7, 10, 15, and 20 robots and measure the mean *order*  $\Phi$  and formation error  $\varepsilon$  to evaluate the scalability of the proposed method. The result in Figure 4.6 shows that as the number of robots increases, the PRC maintains strong consensus among robots with a mean *order* close to 1. The formation error, excluding the formation generation and the transition stage, remains low with variations around 0.05 m. The PRC is therefore scalable with stable performance across different swarm sizes.

#### 4.4.3 Software-in-the-loop verification

We have carried out software-in-the-loop (SIL) tests to evaluate the performance of the proposed controller in practical conditions. The environment is a narrow space that consists of two large obstacles forming a cave-like structure as shown in Figure 4.7a. The



(a) The cave-like environment

(b) The drone model [58], [59]

Figure 4.7: The robot and environment structure used for software-in-the-loop tests.

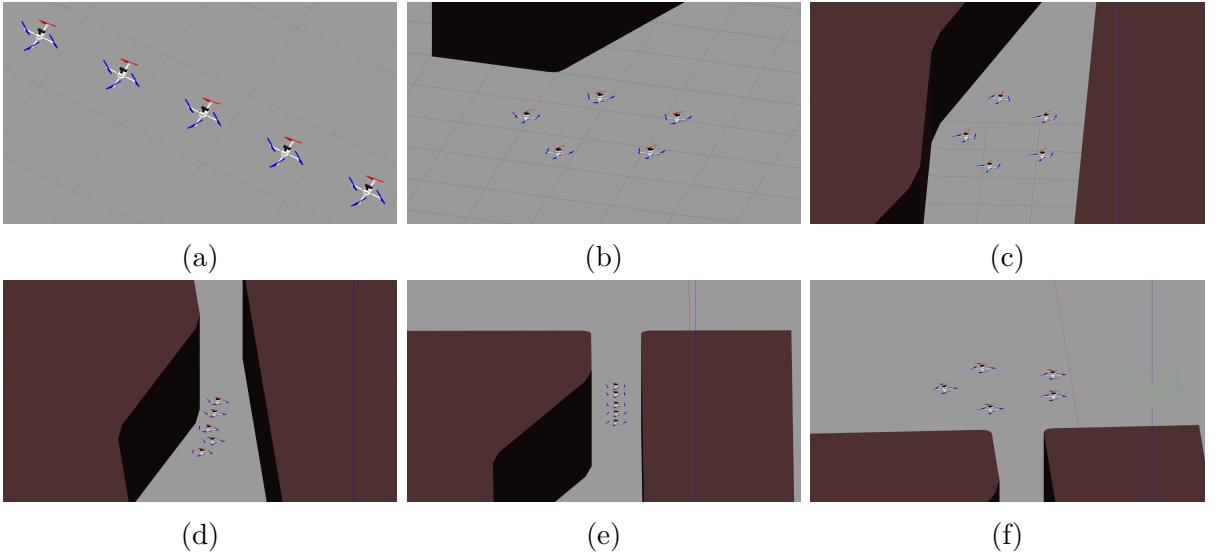


Figure 4.8: Reconfiguration process of the robot swarm in a SIL test: (a) initial positions of the robots; (b) form the desired pentagon shape; (c) shrink the formation in adaption to the environment; (d)-(e) switch to “*tailgating*” mode to travel through the narrow passage; (f) transform back to the desired shape.

robots include five homogeneous Hummingbird quadrotors<sup>1</sup> obtained from the RotorS simulator [58] with an arm length of 0.17 m, a mass of 0.716 kg, the rotor thrust constant of  $1.6^{-2}$  N/A, and the rotor drag constant of  $8.54858 \times 10^{-6}$  Nm/A, as depicted in Figure 4.7b. Each robot is equipped with a range sensor to collect point cloud data of the environment, a positioning module for localization, and a communication module to interact with other robots.

Figure 4.8 presents the formation reconfiguration process as the swarm navigates through the environment. The robots continuously collect data about the environment and based on it adjust their formation to ensure safe operation. Figure 4.9 provides a detailed view of the result. Each UAV determines its mode and desired position based on the perception of the surrounding environment and information about its neighbors. The robots together form the relevant shape in a decentralized manner, as shown in Fig-

<sup>1</sup>Source code used to setup SIL tests in Gazebo - [https://github.com/duynamrcv/hummingbird\\_simulator](https://github.com/duynamrcv/hummingbird_simulator)

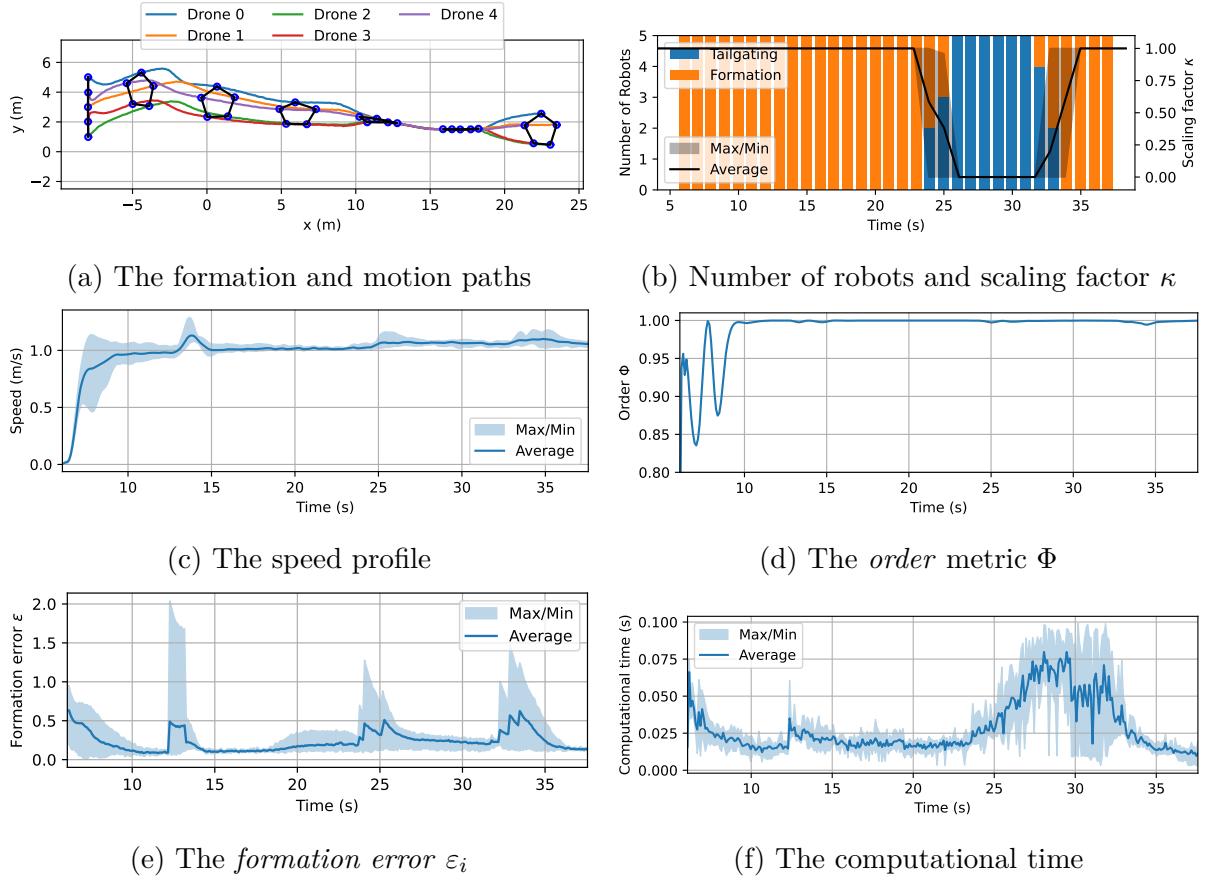


Figure 4.9: Results of the SIL tests

ures 4.9a - 4.9b. Requirements for speed, order, and formation accuracy are also met, as depicted in Figures 4.9c - 4.9e. Moreover, the computational time per controller iteration, shown in Figure 4.9f, indicates that the system can operate at a sampling rate of 10 Hz in the worst-case scenario, which is sufficient for real-time robot operation.

#### 4.4.4 Discussion

Evaluation and comparison results show the key properties of the proposed control method as follows:

##### Decentralization

The PRC is fully decentralized as each robot makes decisions based solely on its own sensor data and information from its neighbors. Unlike the approach in [11], which requires system-wide communication to obtain information on all robots, the PRC utilizes only one-hop communication between the robot and its neighbors to update predictive states.

##### Reliability

The PRC can navigate the robots through complex environments with desirable performance metrics such as low formation error, stable formation direction, and high speed. Unlike previous studies [10]–[12], [64] which only shrink or expand the formation to adapt

to environment variations, the PRC enables the robots to completely transform to a new formation to safely maneuver through tight spaces.

### Scalability

The PRC can operate with different swarm sizes, for example from 4 to 20 robots, without requiring any modifications to the algorithm. It maintains reliable performance and consistent metric values across various swarm sizes.

### Robustness

The PRC provides robustness to operate in different environmental structures. Through real-time data obtained from local sensors and communication networks, the swarm can dynamically expand, shrink, or transform into a line shape to optimize its ability to navigate through narrow spaces.

## 4.5 Conclusion

In this work, we have presented an optimal predictive reconfiguration control method to guide a swarm of robots through cluttered environments with varying path widths. The controller features two modes, “*formation*” and “*tailgating*”, and a scaling factor that enable the swarm to adapt its shape to environmental conditions. A set of cost functions is introduced to enforce formation constraints and safety requirements while enabling state prediction for enhanced control performance. Evaluation results show that the proposed controller effectively navigates the robot swarm through complex environments with narrow passages. The control performance is superior in most evaluation metrics compared to two other state-of-the-art methods. Software-in-the-loop tests further confirm the validity and practicability of the proposed controller.

# Chapter 5

---

## Conclusion and Future Works

### 5.1 Conclusion

Driven by the transformative potential of multi-UAV systems across various missions, such as search and rescue (SAR), this master’s thesis presents a set of contributions addressing the autonomous navigation of UAV formations through clustered environments, particularly in narrow spaces. Below is a summary of the master’s thesis, highlighting the key insights and contributions.

To enable the safe navigation of a V-shaped formation, *Method 1* focuses on formation control by designing a distributed architecture for self-reconfiguration. The proposed method integrates several adaptive behaviors that allow the V-shape formation to shrink or expand based on the width of the environment. In collision-free scenarios, the approach ensures shape maintenance and inter-UAV collision avoidance.

Building upon the foundation laid by *Method 1*, *Method 2* extends multi-robot navigation to support a variety of configurations beyond the V-shape. This work introduces an event-based reconfiguration control, constructed using artificial potential fields to model the required behaviors. The approach includes two modes, “*formation*” and “*tailgating*”, to guide the UAVs and adapt their shape dynamically according to environmental constraints. The stability of the proposed method is rigorously proven using the Lyapunov theory. This approach enables the robot formation to navigate safely through narrow spaces by adjusting its scale or transforming into a straight-line configuration when necessary.

While the first two methods provide foundational strategies for navigation, they do not explicitly account for the physical constraints and limitations of robotic systems. To address this gap, *Method 3* formulates the reconfiguration formation control problem as an optimization problem, fully incorporating system constraints. The behaviors introduced in *Method 2* are converted into objective functions, with control signals derived by minimizing a weighted-sum cost function subject to system constraints. Environmental data is collected to estimate the width of the surrounding space, ensuring the optimal solution to enhance the effectiveness of reconfiguration formation control, and prioritizing safety and collision avoidance.

## 5.2 Future direction

Future research will focus on developing safety-critical reconfiguration control strategies to enhance the safe motion of UAVs and minimize the risk of collisions with other robots or environmental obstacles. Given the challenges posed by narrow spaces, mitigating collisions with neighboring robots or the environment is crucial. Enhancing safety-critical mechanisms is essential to ensure reliable, collision-free operation.

Additionally, exploring non-communication formation reconfiguration control methods will be key to improving system autonomy and robustness. Vision-based perception is a promising alternative for addressing these challenges, providing the system with the capability to perceive and adapt to its surroundings independently. These advancements aim to push the boundaries of autonomous UAV formations, ensuring they can operate safely and efficiently in complex and constrained environments.

---

# List of related publications

## Articles

**Duy-Nam Bui**, Manh Duong Phung, and Hung Pham Duy. “Event-based Reconfiguration Control for Time-varying Formation of Robot Swarms in Narrow Spaces,” *Manuscript*.

**Duy-Nam Bui**, Manh Duong Phung, and Hung Pham Duy. “Predictive Reconfiguration Control for Multi-Robot Formation in Cluttered Environments,” *Manuscript*.

**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.

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.

**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**, 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**, 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 (ICMRI)*, Da Nang, Vietnam, 2024.

## Open-source software

UAVs simulator – [https://github.com/duynamrcv/hummingbird\\_simulator](https://github.com/duynamrcv/hummingbird_simulator).

NMPC controller – [https://github.com/duynamrcv/hummingbird\\_nmpc](https://github.com/duynamrcv/hummingbird_nmpc).

SRVC strategy – [https://github.com/duynamrcv/reconfigurable\\_vshape](https://github.com/duynamrcv/reconfigurable_vshape).

ERC strategy – <https://github.com/duynamrcv/erc>.

PRC strategy – <https://github.com/duynamrcv/prc>.

---

## 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](https://doi.org/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](https://doi.org/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](https://doi.org/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](https://doi.org/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](https://doi.org/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](https://doi.org/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](https://doi.org/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](https://doi.org/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. 139176–139185, 2021. DOI: [10.1109/ACCESS.2021.3119232](https://doi.org/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](https://doi.org/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. Virág, 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. 106332, 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. 105398, 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](https://doi.org/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](https://doi.org/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](https://doi.org/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](https://doi.org/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](https://doi.org/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](https://doi.org/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](https://doi.org/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](https://doi.org/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](https://doi.org/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](https://doi.org/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](https://doi.org/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](https://doi.org/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.
- [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.
- [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.
- [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. 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. 10529–10535. DOI: 10.1109/IROS47612.2022.9981858.

- [48] 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](https://doi.org/10.1109/ICRA48506.2021.9561902).
- [49] 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](https://doi.org/10.1109/TCYB.2019.2913326).
- [50] 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](https://doi.org/10.1007/s10514-020-09913-0).
- [51] 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](https://doi.org/10.1177/0278364917719333).
- [52] 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](https://doi.org/10.1016/j.robot.2012.10.009).
- [53] 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](https://doi.org/10.1109/iros.2018.8594438).
- [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](https://doi.org/10.1016/j.robot.2017.07.007).
- [55] 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](https://doi.org/10.1016/j.ast.2020.105981).
- [56] N. Xuan-Mung and S. K. Hong, “Robust adaptive formation control of quadcopters based on a leader–follower approach,” *International Journal of Advanced Robotic Systems*, vol. 16, no. 4, p. 1729 881 419 862 733, 2019. DOI: [10.1177/1729881419862733](https://doi.org/10.1177/1729881419862733).
- [57] 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](https://doi.org/10.1109/TITS.2021.3107336).
- [58] 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](https://doi.org/10.1007/978-3-319-26054-9_23).

- [59] 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.
- [60] 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.
- [61] T. Nguyen, H. M. La, T. D. Le, and M. Jafari, “Formation control and obstacle avoidance of multiple rectangular agents with limited communication ranges,” *IEEE Transactions on Control of Network Systems*, vol. 4, no. 4, pp. 680–691, 2017. DOI: 10.1109/TCNS.2016.2542978.
- [62] T. Zhang, D. Dong, Z. Du, *et al.*, “Swarm control based on artificial potential field method with predicted state and input threshold,” *Engineering Applications of Artificial Intelligence*, vol. 125, p. 106567, Oct. 2023, ISSN: 0952-1976. DOI: 10.1016/j.engappai.2023.106567.
- [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] 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.
- [67] H. Liu, Y. Wang, F. L. Lewis, and K. P. Valavanis, “Robust formation tracking control for multiple quadrotors subject to switching topologies,” *IEEE Transactions on Control of Network Systems*, vol. 7, no. 3, pp. 1319–1329, 2020. DOI: 10.1109/TCNS.2020.2976271.
- [68] 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.
- [69] 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.
- [70] 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.

- [71] 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](https://doi.org/10.1038/s41592-019-0686-2).
- [72] D. Kraft, *A Software Package for Sequential Quadratic Programming* (Deutsche Forschungs- und Versuchsanstalt für Luft- und Raumfahrt Köln: Forschungsbericht). Wiss. Berichtswesen d. DFVLR, 1988.
- [73] 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. 102668, Jan. 2021, ISSN: 1361-9209. DOI: [10.1016/j.trd.2020.102668](https://doi.org/10.1016/j.trd.2020.102668).