Formation Control of Localised and Decentralised Robotic Swarms

Hongjun Yu

September 6, 2017

Thesis submitted for the degree of
Doctor of Philosophy
in
Electrical Engineering
at The University of Adelaide
Faculty of Engineering, Computer and Mathematical Sciences
School of Electrical and Electronic Engineering

SUB CRUCE LUMEN

THE UNIVERSITY
of ADELAIDE

Contents

List of Tables

List of Figures

List of Symbols

List of Symbols in Chapter 2

x_i	the coordinate of robot i
u_i/u_{x_i}	control force of robot i
g_i/N_i	set of all nearby robots of robot x_i
x/X	a team of robots in set form
$\mathbf{X}$	a team of robots in matrix form
D	the objective map/formation in set form
$\mathbf{D}$	the objective map/formation in matrix form
$\vec{D}_{d_i}/\bar{g}_i$	a set of displacements from the objective map
R	sensing range
$G_{x_i}(R)$	a group of neighbours under sensing range R
$d^{x_i}(G_{x_i}(R))$	the private local map formation of $G_{x_i}(R)$
$d(G_{x_i}(R))$	local map formation by robot i $'$s neighbours in set form
$\mathbf{d}(G_{x_i}(R))$	local map formation by robot i $'$s neighbours in matrix form
$d(X)$	the map formation of the robot team
r	limited control range
$\mathcal{G}_{x_i}(R, r)$	control neighbourhood set of robot x_i
$\mathcal{G}(R, r)$	control neighbourhood set of robot team
$L(X)$	Laplacian matrix
$D_{d'}(R)$	map robot ds neighbours under sensing range R
W_{x_i}	set in Conflict Checking
X'_{x_i}/X''_{x_i}	set in Conflict Locating
$I_{ne}(x'')$	evaluation criterion in Conflict Locating
$\mathcal{M}_{x_i}(t)$	memory set at t in Definition 1
$\mathcal{E}_{x_i}(t)$	set in Conflict Resolving

$J_{\mathcal{N}(r)}$	mapping error under set $\mathcal{N}(r)$ and control range r
$J^{d}_{x_i}$	assessment difference between robot x_i and robot team
T_{x_i}	mapping misalignment between $\mathcal{G}_{x_i}(R,r)$ and $d^{x_i}(G_{x_i}(R))$
T	overall mapping misalignment
$\mathcal{K}_{G_{x_i}(R)}(t)/\mathcal{H}_{d_i}(D')$	set in Protocol 2

List of Symbols in Chapter 3

F	objective formation
s^i_ι/s_i	group of communal neighbours/the collective
m^i_j	communal group reference
w^i_j	set of s^i_ι that can access the same m^i_j
$\Upsilon/\Delta\Upsilon/\Pi$	limited displacement formation
τ	sampling period
$d^{s^i_\iota}(x')$	mapping decision of x' in F
$d(s^i_\iota)$	mapping decision of s^i_ι in F
$\delta_{s^i_{\iota,l}}(k)$	control edge from robot x_i to robot $x^{s^i_\iota}_l$
$\delta(k)/\delta_{s_i}(k)/\delta_{s_i}$	group of control edge
$\bar{\delta}_{s^i_{\iota,l}}(k)$	normalised control edge
$\bar{\delta}(k)/\bar{\delta}_{s_i}(k)/\bar{\delta}_{s_i}$	group of normalised control edge
$L_{X'}$	formation residue energy under communal neighbours X'

List of Symbols in Chapter 4

$\mathcal{N}(\mathbf{X},R,x_i)$	set of all nearby robots of robot x_i
$U^i_k\mathbf{D}$	exclusive mutual neighbour group
U	division operator
$U\mathbf{D}$	mapping source
$\mathcal{E}$	group division operator
$\mathcal{P}$	mapping protocol
$\mathcal{C}$	control law
$d^{i,k}(x_i)$	point that x_i is mapped to
$\phi^{i,k}_j/\phi^{i,k}$	group of control edges/control edge
V^{G^i}	criterion to evaluate the overlapped area around robot x_i
$\mathbf{D}^i$	collection of compact mapping decision sets in $\mathbf{D}$
$J^{G^i_k}$	cross-correlation criterion

List of Symbols in Chapter 5

X	group of N follower robots
Z	group of M leader robots
E_X	orientation vector set
V_l^X	speeds of left wheels for members of X
V_r^X	speeds of right wheels for members of X
V^X	speeds of robot centres for members of X
R_X/R_Z	radiuses for the robot teams X and Z
$W_{z_j}^{x_i}$	angle set for robot x_i from z_j
$\kappa_{z_i}^{x_i}$	desired position for robot x_i from z_j
τ	sampling period
$\theta^{x_i}(k)$	small angular displacement at instant k
$e_{x_i}(k)$	robot x_i's orientation
$r_l^d(l)$	desired trajectory radiuses of the left wheel
$r_r^d(k)$	desired trajectory radiuses of the right wheel
$r^d(k)$	desired trajectory radiuses of the robot center
$r_l^{x_i}(l)$	x_i's desired trajectory radiuses of the left wheel
$r_r^{x_i}(k)$	x_i's desired trajectory radiuses of the right wheel
$r^{x_i}(k)$	x_i's desired trajectory radiuses of the robot center

Abstract

Robot swarms consist of multiple autonomous robots, which detect and interact with their local environments. The fundamental intelligence is observed when a chaotic swarm reaches a stable and orderly objective formation. The process is important because the objective formation is designed such that the swarm achieves more than the sum of its individuals.

A formation is a set of positions or tasks, and intelligent swarms are capable of self-organising and task allocating. Given an objective formation, individuals of a swarm reach different objective positions and perform different tasks. This implies task allocation in different possible choices. For each individual, the path to its objective position is regarded as the effort to take, and the inclination to different objective tasks means different efforts. The challenge is that it needs to choose wisely in the interaction with its neighbourhood. Changes of choices are compromises and each progress to the objective position imposes influence on its neighbourhood. The collective intelligence comes from series of individual decisions in the process.

In this thesis, we consider four problems that arise with the challenge. We use techniques from graph theory and agent-based design to address them.

Formation control algorithms should not impose heavy burden in the communication network. Thus, to start with, limited sensing and communication are assumed, and the robots have minimal access to each other's identity through locally established channels. The control strategy is proposed based on local optimisation and multi-object mapping for a team of robots. Robots are able to make mapping decisions based on local information. To achieve the local optimal mapping decisions for each robot, two novel multi-object mapping protocols are designed. The first protocol performs conflict locating and resolving, and the second adopts a most-neighbour mapping strategy.

The formation problem is further addressed for a scalable team of robots subject to limited sensing with no communication. The robots themselves are fully independent with no designated roles. Scalable objective formation design is proposed such that the robot formation is scalable. Under the assumption that the data transmission among the robots is not available, a novel controller and a protocol are designed that do not rely on communication. As the controller only drives the robots to a partially desired formation, a distributed coordination protocol is proposed to resolve the imperfections.

The case is investigated where the objective formations are arbitrary and have fixed sizes. Multi-objective mapping is proposed for the individual robots to identify their positions in the objective formation. The fixed formation size induces mapping loops, and to avoid local optimum traps, an evaluation method imposes a weak restriction on the predefined formation, rendering it almost arbitrary. To enhance the robustness, the minimal local topology is proposed, and to reduce the computation burden and avoid the infinite trajectory loop, the coordination protocol is modified by introducing probability.

The practical problem of collision avoidance is also studied. The leader-follower scheme is implemented on a multi-robot platform. On the premise of coordinated control laws, globally desired formation is achieved. The same problem in the path-planning perspective is considered on a global scale. Disc obstacles are filtered and clusters are identified based on their intersections. The path planning algorithm is designed based on obstacle clusters.

Keywords: formation control, limited sensing range, no communication, collision avoidance, path planning.

HDR Thesis Declaration

I certify that this work contains no material which has been accepted for the award of any other degree or diploma in my name, in any university or other tertiary institution and, to the best of my knowledge and belief, contains no material previously published or written by another person, except where due reference has been made in the text. In addition, I certify that no part of this work will, in the future, be used in a submission in my name, for any other degree or diploma in any university or other tertiary institution without the prior approval of the University of Adelaide and where applicable, any partner institution responsible for the joint-award of this degree.

I give consent to this copy of my thesis when deposited in the University Library, being made available for loan and photocopying, subject to the provisions of the Copyright Act 1968.

I acknowledge that copyright of published works contained within this thesis resides with the copyright holder(s) of those works.

I also give permission for the digital version of my thesis to be made available on the web, via the Universitys digital research repository, the Library Search and also through web search engines, unless permission has been granted by the University to restrict access for a period of time.

Signed: . . . . . . Date: . . . September 6, 2017 . . .

Acknowledgements

In the course of writing this thesis, my supervisors Professor Peng Shi and Professor Cheng-Chew Lim have offered their continuous support, from choosing the research topic to polishing the proposal reports, from writing the outline to settling the technical specifics in each draft. For their patience, guidance, selfless dedication and professionalism, I would like to express my utmost admiration and sincere gratitude.

I gratefully acknowledge School of Electrical and Electronic Engineering for supporting my research financially. Also, I wish to thank Dr Hong Gunn Chew and Mr Danny Giacomo, who provided me enormous technical supports to build the multi-robot platform.

I would like to thank my colleagues Dr Dongzhe Wang, Ms Huiyan Zhang, Mr Zhongrui Hu and Ms Bing Yan. Also, I am thankful to my friends Cheng Zhao, Peng Wang, and for their support and advice. Most importantly, none of this would have been possible without the love and patience of my wife, Ying Wang.

Publications

Journal Articles

Yu, H., Shi, P., & Lim, C. C. (2016). Robot formation control in stealth mode with scalable team size. International Journal of Control, 89(11), 2155-2168.

Yu, H., Shi, P., & Lim, C. C. (2017). Scalable formation control in stealth with limited sensing range. International Journal of Robust and Nonlinear Control, 27(3), 410-433.

Liu, Y., **Yu, H.**, Lim, C. C., & Shi, P. (2017). Steady formation analysis on multi-robot systems. Journal of Control and Decision, 4(1), 12-31.

Yu, H., Shi, P., Lim C. C., & Wu, L. (2017). Formation control based on local optima and multi-object mapping, submitted and under review.

Yu, H., Shi, P., & Lim C. C. (2017). Probability-triggered formation control with limited sensing range, submitted and under review.

Yu, H., Shi, P., Lim C. C., & Wang, D. (2017). Formation control for multi-robot systems with collision avoidance, submitted and under review.

Conference Articles

Yu, H., Shi, P., & Lim, C. C.. Formation control of arbitrary shape with no communication. In Intelligent Control and Automation (WCICA), 2016 12th World Congress on (pp. 356-359). IEEE. 12-15 June 2016. Guilin, China.

Yu, H., Shi, P., & Lim, C. C.. Matching based formation control and analysis of large-scale multi-agent systems. In Systems, Man, and Cybernetics

(SMC), 2015 IEEE International Conference on (pp. 173-178). IEEE. 9-12 October 2015. Kowloon, Hong Kong, China

Yu, H., Shi, P., & Lim, C. C.. Distributed formation control with limited sensing ranges and scalability. In Control Conference (AUCC), 2015 5th Australian (pp. 318-321). IEEE. 5-6 November 2015. Gold Coast, QLD, Australia

Thesis Conventions

Typesetting. This Thesis is typeset using the LATEX2e software. Processed plots and images were generated using Matlab 6.1 (Mathworks Inc.). CorelDRAW 8.4 (Corel Corporation) was used to produce schematic diagrams and other drawings. Pixotec Slicer Dicer (Pixotec Inc.) was used to generate 3D rendered images.

Spelling. Australian English spelling has been adopted throughout, as defined by the Macquarie English Dictionary (A. Delbridge, Ed., Macquarie Library, North Ryde, NSW, Australia, 2001). Where more than one spelling variant is permit- ted such as biassing or biasing and infra-red or infrared the option with the fewest characters has been chosen.

Referencing. The Harvard style is used for referencing and citation in this Thesis.

Chapter 1

Introduction

1.1 Motivations

This thesis focuses on the collective dynamics of artificial swarms. These swarms consist of multiple agents/robots, and the agents are mobile in a physical/virtual formation, detecting and interacting with local environments. The fast growing scale of such swarms (e.g. national transportation in [84, 85, 86], power grid in [87, 88, 89], large-scale economy in [90, 91]) has weakened individual agents abilities and this created a number of challenges. The bottom design on agents is essentially based on egocentrism, like those in [92, 93, 94], but agents free will is restricted by localisation from limited sensing and interacting capacities. Swarms would have switching formations under agents changing minds. There is an overall need on swarms multi-tasking, and thus, top-level design remains to some extent. But it has to be meticulous, as swarms have been decentralised by the free will of their agents.

Conventional formation control methods are insufficient and ill-advised in dealing with decentralisation and localisation, and the difficulties could result in unevenly distributed material and energy, considerable waste of time and labor in system maintenance and upgrade, low efficiency in operation, and system failure. Nature has offered an alternative solution. When a chaotic swarm, such as a cloud of bees, reaches a stable and orderly formation, the fundamental intelligence is established [95, 96, 97, 98, 99]. The process is important because in the orderly formation, the swarm can achieve more than the sum of its agents and cope with multi-tasking. Such formation is a set of positions, and at each position, an agent performs a specified task. As a result, intelligent swarms are capable of self-organising and task allocating. For individual agents, they need to make decisions wisely in the interaction with their neighbours. Collective intelligence comes from a series of agent decisions in the process.

The thesis aims to construct collective intelligence in multi-robot systems, such as multi-vehicle systems in [77, 78, 79, 80, 81, 83]. This is achieved by developing schemes to manipulate the robots motives in the context of local interaction. The schemes are dominantly agent-based and they are well suited to provide solutions in localised and decentralised systems. Inter-robot communication, if any, is assumed to be established on local wireless network. The idea of swarm intelligence is used to solve the data processing problem in large-scale decentralised systems. For example, given physical swarms, the collective intelligence can be used in autopilot for a network of ground, aerial and underwater vehicles. The network is decentralised since the vehicles perform different tasks. The vehicles are localised because they need to operate in a large area and this weakens their detecting and communicating capacities. Our schemes are well suited for such system.

Contribution from our work in this thesis is as follows:

1. Investigate the principles of formation control among smart robots with limited abilities in sensing, communication and computation. The schemes are based on graph theory and involve algorithms on task allocation, inter-agent coordination and interests-conflict resolution.

2. Build a fundamental platform to practice a leader-follower formation manipulation scheme. Programmable Arduino robots are programmed with the ability to avoid collisions with others and intrusion into the frame boarder. It is demonstrated that the follower robots are able to corner the leaders adaptively in a reasonable formation and both groups are scalable.

3. Dynamic programming is used in path planning at the presence of disc obstacles. Based on obstacle intersections, the clusters are identified from each obstacles perspective and agreements among multiple perspectives led to an exhaustive set of possible convex clusters. The optimal path is derived based on the convex clusters.

1.2 Preliminaries

Multi-robot dynamics is complex. The behaviour of a single robot is often predictable since its model is sufficiently/reasonably accurate and its environment evolves in some known way in most cases. Thus, requirements of a single dynamic robot are easy to meet. When it comes to multiple dynamic robots, they can communicate and interact with each other and observability of the environment is limited for any single robot. Predicting robots behaviours becomes a challenge, let alone to drive all robots to reach a common goal, such as a group of helicopters delivering supplies to an area in certain formation, and a group of robots cooperating to accomplish a complex task.

Network complexity has significant influence on system performance. For mobile nodes (vehicles), the goal is to reach an optimal communication structure. To illustrate, consider the network graph shown in Figure 1.1-1. It uses 11 links. Such structure is simpler than that in Figure 1.1-2, where there are 16 links. More links may make the system more robust, but this also implies heavier burden on the network and computation. In this case, the objective is for the nodes to reach a trade-off, and one possible optimal structure can be observed in Figure 1.1-3. For quality of the links, limited communication and sensing ranges, data dropouts and unreliable links should also be considered in practice.

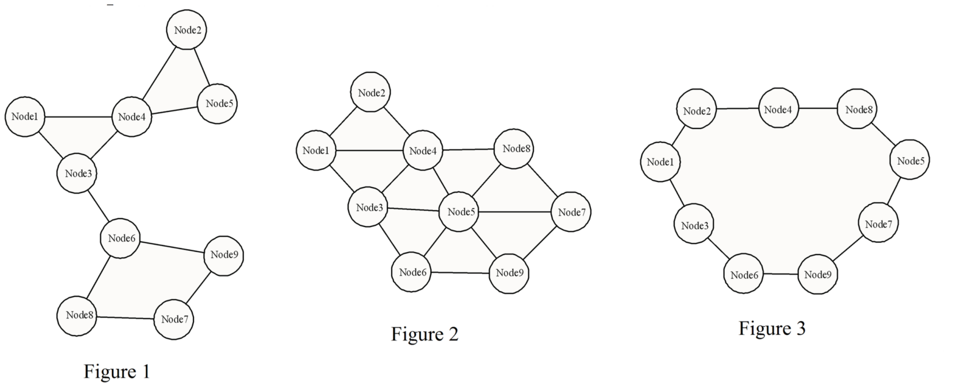


Figure 1.1: Three network structures among nine nodes.

The basic theory of all related methods is the graph theory [100, 101, 102, 103]. It has been proved in many cases [104, 105, 106, 107] that over a connected graph, control force u_i of agent x_i bellow could drive all agents to a consensus:

$$\begin{cases} \dot{x}_i = u_i, \\ u_i = f(N_i, g(N_i)), \end{cases} \tag{1.1}$$

where N_i is a set of all nearby agents of agent x_i, $i = 1, \ldots, n$, and there are n agents in total; f is the control force and g is the map function. Many schemes can be proposed to improve the system performance. Consensus and formation control is one of the main concerns of agent-based system. The agents follow desired trajectory via a time-varying or time-invariant communication graph and reach a user defined formation. How much information agents have poses direct influence on performance of the system, in terms of fuel consumption and wandering time from initiation to finishing a task.

1.3 Literature Review

1.3.1 Theoretical Developments and Applications of Formation Control

Formation control problems are ubiquitous from engineering to social science, including autonomous cooperation in logistics, remote sensing in robotic exploration, coordination in UAV surveillance, and coordination in situation-based relief supplies allocation [9, 23]. Even though an all-to-all topology for communication is desirable, it is often not practical. Sensors may have limited sensing range such as sonar or infrared sensors [10], communication channels may have limited capacity with such devices as bluetooth, and communication may be limited to short distances due to power insufficiency. Interaction topology, measurements and communication are some of the intrinsic problems that can affect the performance of formation control and even make the control system unstable [5, 16, 17, 21, 24, 25, 27, 33, 38]. Research has been done to improve the performance under these circumstances.

The problem of formation coordination with range-only measurements is investigated in [31, 123]. Each agent took one action at a time, rendering formation not sufficiently agile. Furthermore, a global clock and some global information may be needed to coordinate the behaviours of agents. For limited sensing range problems, a potential field is developed in [7] where only desired points are stable. However, the desired points have constraints and extra information storage and transmission for potential field are needed. A relative distance based guaranteed-cost controller was proposed in[10], for which only line formation is considered. A vector field-based method was used in [12] used with a free sensing range. Although this shows improved flexibility, a hierarchical structure is taken and agents are assumed to take different roles. Tags were used in [1, 4, 6] to maintain unconstrained formation but the algorithms were too descriptive, rather than being theoretical, to provide insights about this method.

The network coordination problem is studied [39] with an inter-agent relative position based Laplacian technique. Here, the agents are hierarchical, established links are invariant, and when scale of the system grows, computation burden may increase exponentially. A class of distance-based formations is defined in [35]. By assuming that the formation graph is a tree, a control law based on negative gradient and potential field was proposed and the relationship between the system null space and the tree graphs was established in [34]. However, as global information must be used in the method, the application circumstances became rather narrow. The stability margin for maintaining the desired formation is studied in [36], and it is found that introducing small asymmetry could improve the system performance in the higher dimensional information graph. Their findings of some intrinsic properties in the multi-agent systems are significant. Nevertheless, there are still important issues to address regarding the maintenance of the system connectivity and the achieving of the desired formation, which is the motivation of this work.

Some researchers focused on limited sensing problems. The formation control problem with collision avoidance based on a consensus strategy was studied in [40]. The formation shape and the collision avoidance were incorporated into the weighted graphs. The sensing range was assumed to be limited, as was the minimal avoidance range. The control method was based on the framework of an artificial field. However, the controller allows agents to access the centroid of the robot formation, making some global information available. In addition, attempts are made to reach all-to-all topology by constructing a special data communication protocol. The coverage control for a multi-robot network was investigated in [43]. It was assumed that the sensors only had anisotropic views and that the data communication range was limited, though global communication could still be achieved by designing an appropriate protocol. Under such a problem setup, the position topology is fixed and the robots need to establish another layer of sensor topology. Other than maximising data collection, the optimal coverage could also benefit the motion control of the robots. Related results can be seen in [108, 109, 110]

Formation shapes and topology based schemes have also attracted a lot of interest [124, 125]. The flexible scale control problem was investigated by imposing shape constraints in [42]. A criterion was proposed based on the geometries' degrees of similarity with respect to the desired shape. However, the method was applied under a fixed topology, and as each agent had the access to the global coordinates, the decentralisation of the system was compromised. The stability and robustness of large-platoon vehicles was analysed under nearest neighbour interaction in [41]. The predecessor-following and the symmetric bidirectional architectures were studied, the former of which had a faster convergence rate but with a higher algebraic growth of initial errors. The key of both techniques featured a selection of appropriate fixed topology from a group of permissible topologies but eventually, the proposed scheme only worked under a fixed selected topology.

Cooperation among the robots is investigated under different scenarios on the sensing. One interesting study was a practical application of formation control for a team of Arduino cars in [49]. Given the power and accuracy limitations in the sensors, the range of displacement measurements was reduced. The same was assumed with the data communication range, and the Arduino robots were fully distributed. Stability was achieved on the limited range sensor model but the control method only worked for a string formation. A group of non-holonomic robots were categorised as either leaders or followers in [54]. Active vision based on the followers' pan-controlled cameras was used to provide measurements of the leaders, and no inter-robot range estimation was used. Cameras equipped on multiple unmanned aerial vehicles (UAVs) were used to gather local measurements for groups of robots in [51]. Subsets of the robots corresponded to the individual views of the UAVs. Both of the two sensing techniques above lead to the localisation around the robots.

There are some other situations where localisations apply. From communication point of view, the nonholonomic-wheeled robots were subject to localisation in [59] to maintain the connectivity. The difficulty here is that the controller needed global information, which involved the positions of all other robots. Limited viewing range was assumed for the individual robots in [55]. Only gathering and chain formation were compared in the decentralisation scheme with those in the centralisation scheme. Some localisation was induced by the control methods. In the leader-follower approach in [57], the followers only received information from the intermediary leader, who had the information of the reference trajectory. Moreover, decentralisation could take its form in algorithms in [52], where agents performed under a directed and fixed communication topology. These distributed algorithms are able to solve problems where a global algorithm in [56, 60] could not apply.

The problem of deriving optimal formations is studied intensively. Neural network was used in [62] to find the optimal formation, where the robots were regarded as equal with exchangeable positions. Such a method could be applied to a large-scale system due to the parallel computation. Simulated annealing, genetic algorithm and particle swarm optimisation were tested in [50] to find the optimal formation, and specifically a team of wheeled robots were assigned to deliver an object along a certain path. Multiple leader candidate approach was also used to find the most suitable leader in [13]. Such an on-line method was robust against possible failures of leaders and followers.

1.3.2 Recent Research on Path Planning

Robots often work in complex and uncertain environments. This creates a critical need for path planning algorithms. Guided by these algorithms, robots are able to perform different tasks and be efficient in completing missions. Our work aims to explore the possibility of responsive algorithms such that robots can operate in a congested area.

Various researches have been conducted to solve many practical problems. Experience is often made use of in the investigations. Based on the path library, a retrieval scheme is proposed in [63] to learn from experience. It is proved to provide satisfactory performance, but it tends to rely on existing pattern among different paths and the correlation between new paths and old paths. The adaptive binary searching is adopted in [64] to take advantage of its high efficiency, and previous information is also critical in providing useful reference. A modified A star algorithm is designed in [68] and its effectiveness depends on the additional information that the heuristic function provides. Another improvement is made on A star gird searching in [70] and it is derived from the known multiple basis functions. The navigation in [69] is only based on partial environment knowledge and coarse prior knowledge, but its needs the global plans to some extent since these plans are disintegrated to provide the local plans. Other A-star-based path planning methods can be seen in [111, 112, 113, 114, 115].

A number of on-line and off-line path planning algorithms have been proposed in the past decades [74]. Many studies aimed to achieve real-time response. Practical real-time path planning was proposed in [65], although the solution provided is quasi-optimal. A responsive algorithm is proposed in [72], but multiple restrictions are required and the potential field may only be able to yield quasi optimisation. To combat accidents and traffic densities, highest priority is given to congestion in [73], and this narrows its potential applications. Under compromised optimality, the path planning in [75] can give smooth paths, similar to [76], where artificial potential field is used to approximate obstacles.

Swarm intelligence and genetic algorithms are also introduced to provide mainly off-line path planning [116, 117, 118, 119, 120, 121, 122]. Genetic algorithm and particle swarm optimisation are combined in [119] to solve the problem of local optima, and the response is not fast enough. Dynamic planning is used to improve genetic algorithm in [67], and repetitive computation may be introduced since it is based on grid searching. Multi-objective particle swarm optimisation is studied in [74] to deal with limited sensing and uncertain environment, and the simulations are limited to convex obstacles.

1.4 Structure of Thesis

The rest of this thesis is organised as follows. In Chapter 2, we present formation control to coordinate multiple robots under limited sensing and communication. In Chapter 3, we present coordination schemes to incorporate a growing number of robots, subject to limited sensing and no communication. Chapter 4 presents our solution to strategies to achieve arbitrary formation in a similar setup with that in Chapter 3. Chapter 5 presents the leader-follower formation control implemented on Arduino robots. In Chapter 6, we present path planning algorithms in congested areas. Chapter 7 summarises the research findings and concludes the thesis.

Chapter 2

Formation with Short Sensing Range and Limited Communication

This chapter is concerned with formation control based on local optimisation and multi-object mapping for a team of robots. A limited sensing range is assumed, within which the robots are able to access each other through channels of limited communication. Thus, robots are able to make mapping decisions based on local information. To achieve the local optimal mapping decisions for each robot, two novel multi-object mapping protocols are designed. The first protocol performs conflict locating and resolving, and the second adopts a most-neighbour mapping strategy. As the information flow range of the individual robot is limited, the computational load is upper bounded by the distributed local optimisation. In conjunction with the protocols, an inequality condition is derived to guarantee the connectivity. Numerical examples and analysis are presented to validate the proposed techniques.

2.1 Introduction

This chapter investigates the formation control for a team of robots. Robots are equipped with displacement sensors of limited ranges, and these sensors can provide the displacements among the in-range robots. The only data communication among the in-range robots is sending and receiving the individual identification numbers. Also, robots are assumed to be equally functional and have no hierarchical designated roles, in contrast with [2, 11, 12, 13, 19, 20, 22, 28]. The major components of this chapter consist of conflict resolving solutions based on mapping concepts, connectivity analysis based on local and global optima, and controller and protocol design based on topology and limited sensing and communication.

The goal is to reach a desired formation derived from the objective map. Such constant objective map is predefined and serves as a reference, and all the robots store the information. Based on the measurements from displacement sensors, data received from communication channels, and the objective map, each robot finds its suitable position in the objective map by temporal sequential mapping decisions. Robots make the mapping decisions through protocols, which become available to their neighbours through the communication channels. The challenge is that only local measurements and data are accessible due to limited sensing and communication, similar with those in [3, 7, 8, 10, 14, 22, 30]. Previous methods have used tags [1, 4, 6], potential fields and constrained desirable formations [3, 7, 8, 10, 17]. In this chapter, we investigate the use of tags in our design and theoretical analysis, loosen the constraints on desired formations, and extend the desired formation to almost arbitrary ones.

There are several contributions in this study. Instead of assuming a hierarchical structure and potential field storage [7], all robots are assumed to equally share the same control law, protocol and objective map. The only required data transmission of a robot is sending and receiving the identification numbers to and from others inside its sensing range. Instead of using a complicated potential field [8], a simple controller is designed that can drive the current robot formation to the current map formation. Based on mapping concepts, two original protocols are proposed to transform the current map formation to the objective map while maintaining the formation connectivity subject to a condition. The first protocol uses conflict locating and resolving, and the second protocol uses a most-neighbour mapping scheme to shorten the path to the desired formation. Both methods are shown to be able to drive the robots to a desired formation.

Notations. The notations used are standard. The superscript "T" stands for matrix transposition; $|\cdot|$ denotes the length of a vector; the notation $P > 0$ means that P is real positive definite; $\mathbf{R}^m$ is an m-th dimensional space of real numbers. For operations defined on groups $A = \{a_1, a_2, \dots\}$, $B = \{b_1, b_2, \dots\}$, we have $C = A \oplus B = \{c_1, c_2, \dots\}$, where $c \in A$ or $c \in B$ for any $c \in C$; $C = A \ominus B = \{c_1, c_2, \dots\}$, where $c \in A$ and $c \notin B$ for any $c \in C$; $C = A \odot B = \{c_1, c_2, \dots\}$, where $c \in A$ and $c \in B$ for any $c \in C$; for operations defined on matrices, they follow the common definitions. Denote $a \in A$ if a member a belongs to a set A; $B \subseteq A$ if B is a subset of A by $B \ominus A = \emptyset$. The bold and the non-bold fonts of a symbol denote different variables; the italic and the non-italic fonts of a symbol denote the same variable. Matrices, if their dimensions are not explicitly stated, are assumed to be compatible for algebraic operations.

2.2 Model Description and Controller Design

In our formation control problem, a team of robots

$$\begin{array}{l} X = \left\{ \begin{array}{cccc} x_1 & x_2 & \dots & x_n \end{array} \right\} \text{ in set form, or} \\ \mathbf{X} = \left[\begin{array}{cccc} x_1 & x_2 & \dots & x_n \end{array} \right] \text{ in matrix form} \end{array}$$

is called upon to reach a desired formation from random initial positions, where $x' \in \mathbf{R}^2$ for any $x' \in X$. X is also called a robot formation. Each robot $x_i \in X$ is modelled in the first-order dynamic below.

$$\dot{x}_i = u_{x_i}, \tag{2.1}$$

where input u_{x_i} is the local control on robot x_i.

For performing the team formation, each robot is taken as having the ability to access the objective map

$$\begin{array}{l} D = \left\{ \begin{array}{cccc} d_1 & d_2 & \dots & d_n \end{array} \right\} \text{ in set form, or} \\ \mathbf{D} = \left[\begin{array}{cccc} d_1 & d_2 & \dots & d_n \end{array} \right] \text{ in matrix form,} \end{array} \tag{2.2}$$

where $d' \in \mathbf{R}^2$ for any $d' \in D$. For any $d_i \in D$, we define a set of displacement below.

$$\vec{D}_{d_i} = \left\{ \begin{array}{ccc} d_1^{\vec{D}_{d_i}} & d_2^{\vec{D}_{d_i}} & \dots \end{array} \right\},$$

where $d' - d_i \in \vec{D}_{d_i}$ for any $d' \in D$. The objective of the formation control we concern is to drive the robots to the relative displacements from the objective map D rather than fixed points in space. This means that for any $x', x'' \in X$, if there exists $d' \in \vec{D}_{d^{x'}(x'')}$ such that $x'' - x' = d' \neq 0$, then the robot team X has reached a desired formation. We need to find out what is available to each robot. With the accessible information, the controller on each robot can generate a momentum to drive the robot towards its desired position.

Each robot is equipped with displacement sensors of limited range R. Based on X and R, robot $x_i \in X$ could have a group $G_{x_i}(R)$ of neighbours below.

$$\begin{aligned} G(R) &= \left\{ \begin{array}{cccc} G_{x_1}(R) & G_{x2}(R) & \ldots & G_{x_n}(R) \end{array} \right\}, \\ G_{x_i}(R) &= \left\{ \begin{array}{ccc} x_1^{G_{x_i}(R)} & x_2^{G_{x_i}(R)} & \ldots \end{array} \right\}, \end{aligned} \tag{2.3}$$

where for any $x' \in X$, $x' \in G_{x_i}(R)$ if $|x' - x_i| \leq R$. Even though there are coordinates in $G_{x_i}(R)$, the controller design and the protocol design below are only based on $\bar{G}_{x_i}(R)$.

$$\bar{G}_{x_i}(R) = \left\{ \begin{array}{ccc} x_1^{\bar{G}_{x_i}(R)} & x_2^{\bar{G}_{x_i}(R)} & \ldots \end{array} \right\},$$

where $x' - x_i \in \bar{G}_{x_i}(R)$ for any $x' \in G_{x_i}(R)$. There are only displacements in $\bar{G}_{x_i}(R)$, with which the displacement sensors are able to provide robot x_i.

As there is no central robot, each robot makes its own mapping decisions. Based on $G_{x_i}(R)$ in (2.3) and D in (2.2), any robot x_i can map $G_{x_i}(R)$ to its private local map formation $d^{x_i}(G_{x_i}(R))$ below.

$$d^{x_i}(G_{x_i}(R)) = \left\{ \begin{array}{ccc} d^{x_i}(x_1^{G_{x_i}(R)}) & d^{x_i}(x_2^{G_{x_i}(R)}) & \ldots \end{array} \right\} \subseteq D, \tag{2.4}$$

where $d^{x_i}(x') \in d^{x_i}(G_{x_i}(R))$ for any $x' \in G_{x_i}(R)$. As the communication has minimal flow bandwidth and no complex data structure can be built, any robot x' only perceives the received data from an in-range robot x'' as $d^{x''}(x'')$, and in return, only sends $d^{x'}(x')$ to robot x''. It can be seen in $d^{x_i}(G_{x_i}(R))$ that $d^{x_i}(G_{x_i}(R)) \ominus \{d^{x_i}(x_i)\}$ is a set of mapping decisions that cannot be sent to robot x_i's neighbours.

The private local map formation $d^{x_i}(G_{x_i}(R))$ is a mapping decision on the neighbours made by robot x_i. Through the channels of limited data communication, robot x_i send $d^{x_i}(x_i)$ to its neighbours, and also receive and collect a set $d(G_{x_i}(R))$ or a matrix $\mathbf{d}(G_{x_i}(R))$ of local map formation from its neighbours below.

$$
\begin{aligned}
d(G_{x_i}(R)) &= \left\{ \; d^{x_1^{G_{x_i}(R)}}(x_1^{G_{x_i}(R)}) \quad d^{x_2^{G_{x_i}(R)}}(x_2^{G_{x_i}(R)}) \quad \ldots \; \right\} \\
&\quad \text{in set form or} \\
\mathbf{d}(G_{x_i}(R)) &= \left[\; d^{x_1^{G_{x_i}(R)}}(x_1^{G_{x_i}(R)}) \quad d^{x_2^{G_{x_i}(R)}}(x_2^{G_{x_i}(R)}) \quad \ldots \; \right] \\
&\quad \text{in matrix form,} \qquad (2.5)
\end{aligned}
$$

where $d^{x'}(x') \in d(G_{x_i}(R))$ for any $x' \in G_{x_i}(R)$. Based on $G_{x_i}(R)$ and $d(G_{x_i}(R))$, the controller on robot x_i can generate a momentum, under which robot x_i is expected to gradually reach its desired position in the objective map.

By gathering the mapping decisions from the robots, we obtain the map formation $d(X)$ below.

$$
d(X) = \left\{ \; d^{x_1}(x_1) \quad d^{x_2}(x_2) \quad \ldots \quad d^{x_n}(x_n) \; \right\},
$$

where $d(X) \subseteq D$ and $d^{x'}(x') \in d(X)$ for any $x' \in X$. The desired formation for the robot team is derived from the objective map D under certain map formation $d(X)$. If $d(X) = D$, then there exists a one-to-one mapping from the robot formation to the objective map. The mapping has to be one-to-one for the robots to reach a desired formation. Moreover, a point d in D is also called a map robot.

As the robot formation X is under a limited sensing range R, the objective map D and the map formation $d(X)$ are also under a limited control range r. Thus, based on $G_{x_i}(R) \subseteq X$, $d(X)$ and a given r, the control neighbourhood set can be defined below.

$$
\begin{aligned}
\mathcal{G}(R,r) &= \left\{ \; \mathcal{G}_{x_1}(R,r) \quad \mathcal{G}_{x_2}(R,r) \quad \ldots \quad \mathcal{G}_{x_n}(R,r) \; \right\}, \\
\mathcal{G}_{x_i}(R,r) &= \left\{ \; x_1^{\mathcal{G}_{x_i}(R,r)} \quad x_2^{\mathcal{G}_{x_i}(R,r)} \quad \ldots \; \right\}, \\
d(\mathcal{G}_{x_i}(R,r)) &= \left\{ \; d^{x_1^{\mathcal{G}_{x_i}(R,r)}}(x_1^{\mathcal{G}_{x_i}(R,r)}) \quad d^{x_2^{\mathcal{G}_{x_i}(R,r)}}(x_2^{\mathcal{G}_{x_i}(R,r)}) \quad \ldots \; \right\},
\end{aligned}
$$

where

i) for any $x' \in G_{x_i}(R)$, $x' \in \mathcal{G}_{x_i}(R,r)$ if $|d^{x'}(x') - d^{x_i}(x_i)| \leq r$;

ii) $d^{x'}(x') \in d(\mathcal{G}_{x_i}(R,r))$ for any $x' \in \mathcal{G}_{x_i}(R,r)$.

It can be seen that $|x' - x''|$ is the relative distance between robots x' and x'', and $|d^{x'}(x') - d^{x''}(x'')|$ is their map distance.

We have derived the accessible local information $G_{x_i}(R)$, $d^{x_i}(G_{x_i}(R))$, $d(G_{x_i}$ $(R))$, $\mathcal{G}_{x_i}(R,r)$ and $d(\mathcal{G}_{x_i}(R,r))$ for each robot. Based on $\mathcal{G}_{x_i}(R,r)$ and $d(\mathcal{G}_{x_i}(R,r))$, the control of each robot takes the form as

$$u_{x_i} = \sum_{x' \in \mathcal{G}_{x_i}(R,r)} (-x_i + x' + d^{x_i}(x_i) - d^{x'}(x')), \tag{2.6}$$

where for any $x' \in \mathcal{G}_{x_i}(R,r)$, $d^{x'}(x') \in D$ is the map robot that robot x' is mapped to by a suitable protocol in Section 2.3. The state x' consists of the robot position as its state variable. The robot formation reaches a steady state when $u_{x'} = 0$ for any $x' \in X$. To analyse the formation stability, we need to rewrite (2.1) and (2.6).

There exists $P(X)$ such that the transformation from the objective map D to the map formation $d(G_{x_i}(R))$ can be expressed below.

$$\mathbf{d}(G_{x_i}(R)) = \mathbf{D}P(X). \tag{2.7}$$

For notation simplicity, we can substitute (2.2) (2.3), (2.5), (2.6) and (2.7) and rewrite (2.1) in the following form

$$\dot{\mathbf{X}} = \mathbf{X}L(X) - \mathbf{D}P(X)L(X), \tag{2.8}$$

where $L(X)$ is the topology matrix, which is dependent on $\mathcal{G}(R,r)$ and independent of the map formation $d(X)$. Such a Laplacian matrix is obtained below.

$$\begin{aligned} L(X) &= L_{x_1}(\mathcal{G}_{x_1}(R,r)) + L_{x_2}(\mathcal{G}_{x_2}(R,r)) + \cdots \\ &\quad + L_{x_n}(\mathcal{G}_{x_n}(R,r)(R)), \\ L_{x_i}(\mathcal{G}_{x_i}(R,r)) &= L_{x_1}^{x_i} + L_{x_2}^{x_i} + \cdots + L_{x_n}^{x_i}, \end{aligned}$$

where

$$L_{x_j}^{x_i} = \begin{bmatrix} l_{11} & l_{12} & l_{13} & \cdots \\ l_{21} & l_{22} & l_{23} & \cdots \\ l_{31} & l_{32} & l_{33} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \in \mathbf{Z}^{n\times n},$$

$$l_{ab} = \begin{cases} -I, & \text{if } \ a = b = i, i \neq j, x_j \in \mathcal{G}_{x_i}(R,r); \\ I, & \text{if } \ a = j, b = i, i \neq j, x_j \in \mathcal{G}_{x_i}(R,r); \\ 0, & \text{if } \ \text{otherwise}. \end{cases}$$

It can be seen that the topology is important to the formation stability. We derive below what topology leads to a stable robot formation.

The controller design in (2.6) involves $\mathcal{G}_{x_i}(R, r)$, which is established based on X, D, R and r. Thus, we propose three different types of links below.

a) For any $x', x'' \in X$, there exists a link from x' to x''
if $|x' - x''| \leq R$;

b) For any $x', x'' \in X$, there exists a link from x' to x''
if $|d^{x'}(x') - d^{x''}(x'')| \leq r$;

c) For any $x', x'' \in X$, there exists a link from x' to x''
if $|x' - x''| \leq R$ and $|d^{x'}(x') - d^{x''}(x'')| \leq r$;

Based on the three types of links above, we have

i) given any link type of a), b) or c), the topology is
the collection of all the links;

ii) for any $x', x'' \in X$ and given a link type,
if there exist $x'_1, x'_2, \ldots, x'_j \in X$ and
a path of links that contains a link from x' to x'_1,
a link from x'_1 to x'_2, …,
a link from x'_j to x'', then the topology X is connected
under such link type;

iii) for any $x', x'' \in X$, if there exists a type-a) link
from robot x' to x'', then the two robots can access
each other through the communication channel.

To guarantee that the robot formation is stable and the map formation can be produced (the protocols in Section 2.3 have solutions), the following assumption is made.

Assumption 2.2.1. *The topology of the initial robot formation is connected under any of the three link types.*

Note that by Assumption 2.2.1, the initial robot formation is connected in the three types of links. Under the controller in (2.6) and the protocols in Section 2.3, the condition in Section 2.3.3 guarantees that the subsequent robot formations maintain their connectivity in the three types of links. According to the graph theory [20, 21, 24, 25, 38], such connectivity is important because it ensures that the topology matrix $L(X) \leq 0$ and the protocols have solutions, which is essential for the robot team to reach a stable and desired formation. The current robot formation X is driven to its map formation $d(X)$, on the premise that the formation topology is connected and $d(X)$ is fixed. The proof is omitted.

As the robot formation X is dynamic and $G(R)$ may change, it is necessary to maintain the formation connectivity in order to guarantee $L(X) \leq 0$; as $d(X)$ is possibly changing, the protocol needs to be able to provide the transformation where $d(X) \rightarrow D$. Thus, the protocol design question is how to reach a one-to-one mapping between the robot formation X of all the robots and the objective map D by temporal sequential mapping decisions. Since the sensing range and data communication capacity are limited, only local information is accessible to each robot. A control law and a protocol are needed in order to drive the local mappings from the local optima to a global optimum. In the process, the formation connectivity should be maintained.

In Section 2.3.3, we propose a sufficient condition that can maintain the formation connectivity. Thus, we only need to derive a solution that transforms the map formation $d(X)$ into the objective map D. In a team of robots X under a one-to-one mapping, we have $d(X) = D$ through that $d^{x'}(x') \neq d^{x''}(x'')$ for any $x', x'' \in X$ and $x' \neq x''$. Therefore, the task of protocol can be seen to eliminate the conflicts where $d^{x'}(x') = d^{x''}(x'')$ for any $x', x'' \in X$.

The solution presented in this chapter follows the following line of reasoning. First, a robot finds its local optimal mapping by a protocol, which leads to a temporary desired position under the control law in (2.6). However, obtaining a locally optimal solution for each robot does not necessarily mean that these local solutions sum up to be a globally optimal solution. As the robot approaches its desired position, the solution may lose its local optimality and cause conflicts with its neighbours. Then, dealing with the conflicts involves locating and resolving them. Subsequently, a new locally optimal solution is found. Taking into consideration the robots' mapping history and the finite size of the objective map, the one-to-one mapping is reached eventually.

Note that while there are many simple methods to deal with the formation control problem with fixed topology, these methods demand fairly stringent prerequisites, such as that robots have already identified each other, that robots have fixed roles as leaders or followers, and that communication and sensing are effortless. Clearly, when limitations are imposed on sensing range, available capacity for communication and processor power, these methods become not usable. We present two protocols to solve the mapping transformation problem subject to the aforementioned constraints on sensing and communication.

2.3 Multi-object Mapping

We propose two protocols to find the local optimal mapping for each robot. These local optima gradually converge to a global optimum in the dynamic interaction of the robots. Moreover, we show that these two protocols can guarantee the formation connectivity subject to certain conditions (Assumption 2.2.1 and Theorem 2.3.3), which is discussed in Section 2.3.3.

The initial private local map formations $d^{x'}(G_{x'}(R))$ for $x' \in X$ in (2.4) are needed, because only with $d^{x'}(G_{x'}(R))$ and through communication channels, the initial local map formation $d(G_{x'}(R))$ in (2.5) can be obtained, which is needed for subsequent processes in both protocols. Similarly with defining $G_{x_i}(R)$ based on R, we can define a set $D_{d'}$ of map robot d''s neighbours for robot x_i below.

$$D_{d'}(R^{x_i}_{d'}) = \left\{ \begin{array}{ccc} d_1^{D_{d'}(R^{x_i}_{d'})} & d_2^{D_{d'}(R^{x_i}_{d'})} & \dots \end{array} \right\},$$

where for any $d'' \in D$, $d'' \in D_{d'}$ if $|d' - d''| \leq R^{x_i}_{d'}$. For robot x_i, $R^{x_i}_{d'}$ is introduced to bound the size of D_{d_i}. Based on $D_{d'}$, we introduce the initialisation mapping protocol below.

Initialisation: *For robot x_i at $t = t_0$, there exists $R^{x_i}_{d'}$ such that the size of $D_{d'}$ is no smaller than that of $G_{x_i}(R)$ for any $d' \in D$. Thus, the mapping decision*

$$d^{x_i}(G_{x_i}(R)) = \left\{ \begin{array}{ccc} d^{x_i}(x_1^{G_{x_i}(R)}) & d^{x_i}(x_2^{G_{x_i}(R)}) & \dots \end{array} \right\},$$

where $d^{x_i}(x') \in d^{x_i}(G_{x_i}(R))$ for any $x' \in G_{x_i}(R)$, is made from evaluating the objective function

$$d^{x_i}(G_{x_i}(R)) = \arg \min_{d^{x_i}(G_{x_i}(R)) \subseteq D_{d'}} \min_{d' \in D} \sum_{x' \in G_{x_i}(R)}^{x' \neq x_i} \left((x_i - x') - (-d^{x_i}(x_i) - d^{x_i}(x')) \right)^2 . \qquad (2.9)$$

Remark 2.3.1. *Note that by Assumption 2.2.1, the initial topology is connected after Initialisation has provided each robot with a initial mapping decision. Moreover, we require that each robot stores the objective map D as a database reference. Based on D and $R_{d'}^{x_i}$, the initial private local map formation is obtained by (2.9). As is shown below, such database does not necessarily increase computation burden, which is in fact bounded by $R_{d'}^{x_i}$ in (2.9), and proportional to the number of direct neighbours in (2.13) and first indirect map robot neighbours in (2.16) and (2.23). For robot x_i, through the communication channel with neighbor x', it knows $d^{x'}(x')$, and by the displacement sensor, it knows $x' - x_i$. In addition, once a mapping decision is made, robot x_i sends $d^{x_i}(x_i)$ to all its neighbours within its sensing range.*

Remark 2.3.2. *On the one hand, robot x_i performs self-assessment $d^{x_i}(x_i)$ based on information $G_{x_i}(R)$ from its neighbours. On the other hand, similar information $d^{x'}(x')$ that robot x_i receives comes from self-assessments of all its neighbours $x' \in G_{x_i}(R)$. Since robots are making decisions in a distributed manner, computation burden is shared by all the team members. More specifically, the computation burden of individual robot is proportional to n times of the sum of $D_{d'}$'s dimensions for any $d' \in D$.*

2.3.1 Protocol with Conflict Locating and Conflict Resolving

Through Initialisation and data communication channels, robot x_i is provided with an initial local map formation $d(G_{x_i}(R)) \subseteq D$. Such a mapping decision is based on the local optimal solution in (2.9). However, this local solution is insufficient, as the solution may cause mapping conflicts with its neighbours and lose optimality. Locating and resolving of the conflicts are the approach we adopt. In effect, we aim to find from local solutions the global optimal solution for the formation team.

We now present the protocol that performs the functions of: (i) checking for conflict, (ii) locating conflicts, and (iii) resolving conflicts. Under a triggered condition presented below, the protocol (termed Protocol 2.3.1 below) locates the conflicts and resolves them.

Protocol 2.3.1 (i) - Conflict Checking: For robot x_i at $t = t_l > t_0$, based on $G_{x_i}(R)$ and $d(G_{x_i}(R))$, we have

$$W_{x_i} = \left\{ \begin{array}{ccc} x_1^{W_{x_i}} & x_2^{W_{x_i}} & \dots \end{array} \right\}, \tag{2.10}$$

where for any $x' \in G_{x_i}(R)$, $x' \in W_{x_i}$ if $d^{x'}(x') = d^{x_i}(x_i)$. If $W_{x_i} \neq \emptyset$, then activates the Conflict Locating procedure; if $W_{x_i} = \emptyset$, then return to the beginning of Conflict Checking for Protocol 2.3.1.

Protocol 2.3.1 (ii) - Conflict Locating: For robot x_i at $t = t_l > t_0$, its neighbor set $G_{x_i}(R)$ is divided into two groups if $W_{x_i} \neq \emptyset$ in (2.10). The first group X'_{x_i} is a set of neighbouring robots that are mapped to different map robots from that of robot x_i, which is defined below.

$$X'_{x_i} = \left\{ \begin{array}{ccc} x_1^{X'_{x_i}} & x_2^{X'_{x_i}} & \ldots \end{array} \right\}, \tag{2.11}$$

where for any $x' \in G_{x_i}(R)$, $x' \in X'_{x_i}$ if $d^{x'}(x') \neq d^{x_i}(x_i)$. The second group X''_{x_i} includes those neighbouring robots that are mapped to the same map robots with that of robot x_i, which is defined below.

$$X''_{x_i} = \left\{ \begin{array}{ccc} x_1^{X''_{x_i}} & x_2^{X''_{x_i}} & \ldots \end{array} \right\}, \tag{2.12}$$

where for any $x'' \in G_{x_i}(R)$, $x'' \in X''_{x_i}$ if $d^{x''}(x'') \neq d^{x_i}(x_i)$. It can be seen that

$$X'_{x_i} \oplus X''_{x_i} = G_{x_i}(R), \quad X'_{x_i} \odot X''_{x_i} = \emptyset.$$

If $X'_{x_i} = \emptyset$, then return to the beginning of Conflict Checking for Protocol 2.3.1. If $X'_{x_i} \neq \emptyset$, then robot x'' can be found from evaluating,

$$x'' = \arg \min_{x'' \in X''_{x_i}} I_{ne}(x''), \tag{2.13}$$

where

$$I_{ne}(x'') = \sum_{x' \in X'_{x_i}} |(x' - x'') - (d^{x'}(x') - d^{x''}(x''))|_c;$$

for any $x', x'' \in X$, if $d^{x'}(x') = d^{x''}(x'')$, then $|x' - x'' - d^{x'}(x') + d^{x''}(x'')|_c = \alpha$, where α is a fixed large number; if $d^{x'}(x') \neq d^{x''}(x'')$, then $|x' - x'' - d^{x'}(x') + d^{x''}(x'')|_c = |x' - x'' - d^{x'}(x') + d^{x''}(x'')|$. If $(m+1)\alpha \geq A \geq \alpha$, then A contains m numbers of α. If $x'' = x_i$, then return to the beginning of Conflict Checking for Protocol 2.3.1. If $x'' \neq x_i$, then the X'_{x_i} group is enlarged to form

$$X'''_{x_i} = X'_{x_i} \oplus \{x''\}, \tag{2.14}$$

and proceeds to execute Conflict Resolving.

It is necessary to avoid entering into an infinity loop (see Remark 2.3.7). We take the robots' mapping history into consideration and propose Definition 2.3.1.

Definition 2.3.1. *For robot x_i at $t = t_0$, we define its memory set by*

$$\mathcal{M}_{x_i}(t_0) = G_{x_i}(R).$$

At $t = t_l > t_0$ and with an point set D', $\mathcal{M}_{x_i}(t_l)$ can be updated below.

$$\mathcal{M}_{x_i}(t_l) = \mathcal{M}_{x_i}(t_l^-) \oplus D'.$$

where $t_0 < t_1 < \cdots < t_l$; $t_k^- < t_k$ and $t_k - t_k^- < \epsilon$ for any $\epsilon > 0$; $\mathcal{M}_{x_i}(t) = \emptyset$ for $t < t_0$.

Protocol 2.3.1 (iii) - Conflict Resolving: For robot x_i at $t = t_l > t_0$, if $x'' \neq x_i$ in (2.13), its memory set is updated by

$$\mathcal{M}_{x_i}(t_l) = \mathcal{M}_{x_i}(t_l^-) \oplus X'''_{x_i}. \tag{2.15}$$

Then, evaluate the mapping decision

$$d^{x_i}(x_i) = \arg \min_{d^{x_i}(x_i) \in \mathcal{E}_{x_i}(t_l)} \sum_{x''' \in X'''_{x_i}} \Big((x''' - x_i) - (d^{x'''}(x''') - d_{x_i}(x_i)) \Big)^2, \tag{2.16}$$

where

$$\mathcal{E}_{x_i}(t_l) = \left\{ \begin{array}{ccc} d_1^{\mathcal{E}_{x_i}(t_l)} & d_2^{\mathcal{E}_{x_i}(t_l)} & \ldots \end{array} \right\}; \tag{2.17}$$

for any $d' \in D \ominus \mathcal{M}_{x_i}(t_l)$, if there exists $x' \in X'_{x_i}$ such that $|d' - d^{x'}(x')| \leq r$, then $d' \in \mathcal{E}_{x_i}(t_l)$. The identity of the new map robot $d^{x_i}(x_i)$ in (2.16) is sent to the robots in $G_{x_i}(R)$ through the communication channel.

This protocol generates $d^{x_i}(x_i)$ and updates the memory set $\mathcal{M}_{x_i}(t_l)$. Through data communication, robot x_i receives $d^{x'}(x')$ from its neighbor $x' \in G_{x_i}(R)$, and the new mapping decisions form the new local map formation $d(G_{x_i})$. Then, with the new $d(G_{x_i})$ and G_{x_i} from the displacement sensor, the execution sequence returns to the beginning of Conflict Checking for Protocol 2.3.1. It can be seen that the number of α that $I_{ne}(x'')$ in (2.13) contains is non-increasing at each remapping in (2.16). This means that the number of robots that are mapped to the same robots in the robot formation is non-increasing at each time when a new mapping decision $d^{x_i}(x_i)$ is found in (2.16).

Remark 2.3.3. *Note that the Conflict Locating and Conflict Resolving process aims to map all the robots to mutually exclusive points on the objective map through their local optima in (2.16). The set $\mathcal{E}_{x_i}(t_l)$ is introduced so that computation burden is bounded by the sensing range, regardless of the size of the objective map. Connectivity in Assumption 2.2.1 ensures $\mathcal{E}_{x_i}(t_l) \neq \emptyset$. Moreover, the set $\mathcal{M}_{x_i}(t_l)$ is introduced to provide memory for each robot so that the protocol induced infinite circles are avoided.*

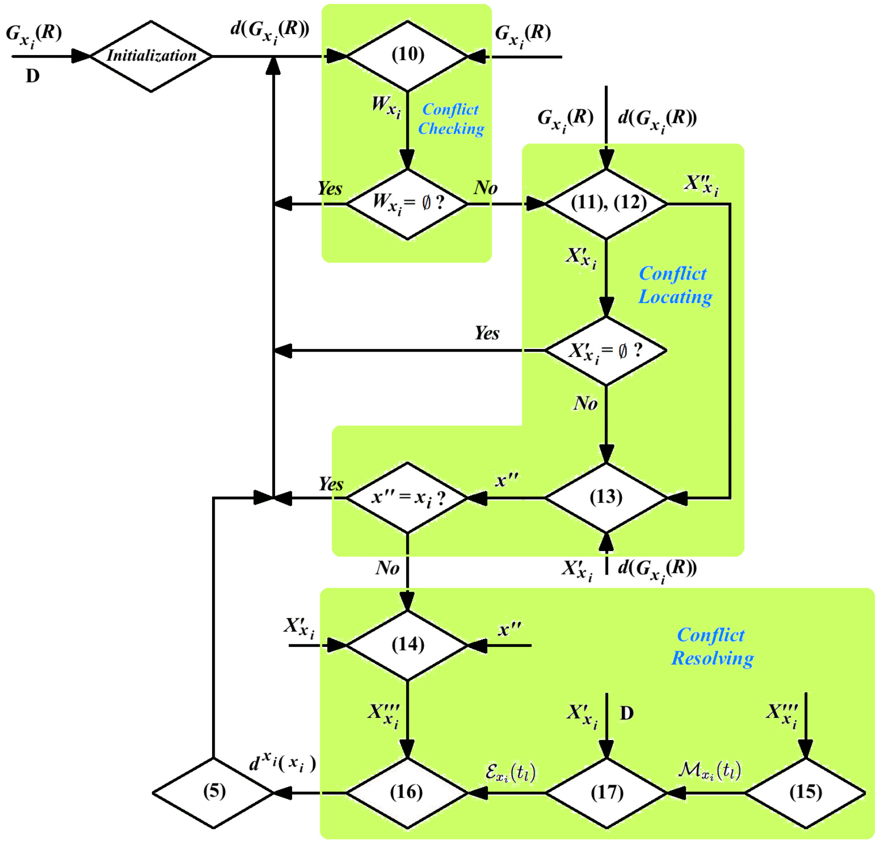


Figure 2.1: Flow chart of Protocol 2.3.1.

We summarise the description of Protocol 2.3.1 as follows.

Protocol 2.3.1. *For any robot x_i at $t = t_l$, after $d(G_{x_i}(R))$ is available through Initialisation and communication and $G_{x_i}(R)$ is available through displacement sensors,* *(a) $\mathcal{M}_{x_i}(t_l)$ and $d^{x_i}(x_i)$ are updated by (2.15) and (2.16)*

in Conflict Resolving if

i) either $W_{x_i} \neq \emptyset$ in (2.10) is found by Conflict Checking, or

ii) $x'' \neq x_i$ in (2.13) is found by Conflict Locating;

(b) return to the beginning of Conflict Checking if

i) either $W_{x_i} = \emptyset$ in (2.10) is found by Conflict Checking, or

ii) $x'' = x_i$ in (2.13) is found by Conflict Locating.

Then, $d^{x_i}(x_i)$ in (2.16) is sent to the robots in $G_{x_i}(R)$ through the communication channel. The Flow Chart of Protocol 2.3.1 is given in Figure 2.1.

Proof of Arrival at a Desired Formation under Protocol 2.3.1

Proof: It can be seen that under a one-to-one mapping and in a connected robot formation, the robots reach a desired formation. Thus, we only need to prove that Protocol 2.3.1 can help the robot team obtain a one-to-one mapping. Under $\mathcal{G}_{x_i}(R, r)$ and $\mathcal{N}_{x_i}(r)$, we define

$$\begin{aligned} J_{\mathcal{G}(R,r)} &= \sum_{i=1}^{n} \sum_{x' \in \mathcal{G}_{x_i}(R,r)} |-x_i + x' + d^{x_i}(x_i) - d^{x'}(x')|_c, \\ J_{\mathcal{N}(r)} &= \sum_{i=1}^{n} \sum_{x' \in \mathcal{N}_{x_i}(r)} |-x_i + x' + d^{x_i}(x_i) - d^{x'}(x')|_c, \end{aligned}$$

where

$$\begin{aligned} \mathcal{N}(r) &= \left\{ \begin{array}{cccc} \mathcal{N}_{x_1}(r) & \mathcal{N}_{x_2}(r) & \ldots & \mathcal{N}_{x_n}(r) \end{array} \right\}, \\ \mathcal{N}_{x_i}(r) &= \left\{ \begin{array}{ccc} x_1^{\mathcal{N}_{x_i}(r)} & x_2^{\mathcal{N}_{x_i}(r)} & \ldots \end{array} \right\}; \end{aligned}$$

for any $x', x'' \in X$, $x' \in \mathcal{N}_{x''}(r)$ if $|d^{x''}(x'') - d^{x'}(x')| \leq r$. Moreover, $n_{d_i}(r)$ is the degree of $D_{d_i}(r)$ for $d_i \in D$. Below, we discuss two situations.

Situation 1: Assume that $\mathcal{G}(R, r)$ is changing. We need to find the boundaries for $J_{\mathcal{G}(R,r)}$. There exists a path from x_i to x_j that contains a link from x_i to x_{j_1}, a link from x_{j_1} to x_{j_2}, ..., a link from x_{j_q} to x_j, where there are q intermediate robots, and $x' \in \mathcal{G}_{x''}(R, r) \in \mathcal{G}(R, r)$ for any link from x' to x''

in the path from x_i to x_j. Thus, we have

$$\begin{aligned}
& |-x_i + x_j + d^{x_i}(x_i) - d^{x_j}(x_j)|_c \\
\leq\ & |-x_i + x_{j_1} + d^{x_i}(x_i) - d^{x_{j_1}}(x_{j_1})|_c \\
& +|-x_{j_1} + x_{j_2} + d^{x_{j_1}}(x_{j_1}) - d^{x_{j_2}}(x_{j_2})|_c \\
& + \ldots + |-x_{j_q} + x_j + d^{x_{j_q}}(x_{j_q}) - d^{x_j}(x_j)|_c \\
=\ & \sum_{l=0}^{q} \left|-x_{j_q} + x_{j_{q+1}} + d^{x_{j_q}}(x_{j_q}) - d^{x_{j_{q+1}}}(x_{j_{q+1}})\right|_c,
\end{aligned}$$

where $x_{j_0} = x_i$, $x_{j_{q+1}} = x_j$. As the formation connectivity is in three levels, we can also have $x_{j_l} \in \mathcal{N}_{x_{j_{l+1}}}(r)$ for $l = 0, \ldots, q$. Thus, it follows that

$$\begin{aligned}
J_{\mathcal{N}(r)} &= \sum_{i=1}^{n} \sum_{x_j \in \mathcal{N}_{x_i}(r)} \left|-x_i + x_j + d^{x_i}(x_i) - d^{x_j}(x_j)\right|_c \\
&\geq \sum_{l=0}^{q} \left|-x_{j_q} + x_{j_{q+1}} + d^{x_{j_q}}(x_{j_q}) - d^{x_{j_{q+1}}}(x_{j_{q+1}})\right|_c.
\end{aligned}$$

Substitute the above inequity into $J_{\mathcal{G}(R,r)}$, we have

$$\begin{aligned}
J_{\mathcal{G}(R,r)} &= \sum_{i=1}^{n} \sum_{x_j \in \mathcal{G}_{x_i}(R,r)} \left|-x_i + x_j + d^{x_i}(x_i) - d^{x_j}(x_j)\right|_c \\
&\leq \sum_{i=1}^{n} \sum_{x_j \in \mathcal{G}_{x_i}(R,r)} \sum_{l=0}^{q} \left|-x_{j_q} + x_{j_{q+1}} + d(x_{j_q}) - d(x_{j_{q+1}})\right|_c \\
&\leq J_{\mathcal{N}(r)} \sum_{i=1}^{n} \sum_{x_j \in \mathcal{G}_{x_i}(R,r)} 1 \\
&\leq J_{\mathcal{N}(r)} \sum_{i=1}^{n} n_{d^{x_i}(x_i)}(r) \\
&\leq n J_{\mathcal{N}(r)} \max_{d' \in D} \{n_{d'}(r)\}.
\end{aligned} \tag{2.18}$$

Similarly, we have

$$\begin{aligned}
J_{\mathcal{G}(R,r)} &= \sum_{i}^{n} \sum_{j \in \mathcal{N}_{x_i}(r)} |-x_i + x_j + d^{x_i}(x_i) - d^{x_j}(x_j)|_c \\
&\geq \sum_{l=0}^{q} |-x_{j_q} + x_{j_{q+1}} + d^{x_{j_q}}(x_{j_q}) - d^{x_{j_{q+1}}}(x_{j_{q+1}})|_c.
\end{aligned}$$

Substitute the above inequality into $J_{\mathcal{N}(r)}$, we have

$$\begin{aligned} J_{\mathcal{N}(r)} &= \sum_{i=1}^{n} \sum_{x_j \in \mathcal{N}_{x_i}(r)} \left|-x_i + x_j + d^{x_i}(x_i) - d^{x_j}(x_j)\right|_c \\ &\leq J_{\mathcal{G}(R,r)} \sum_{i=1}^{n} \sum_{x_j \in \mathcal{N}_{x_i}(r)} 1 \\ &\leq J_{\mathcal{G}(R,r)} \sum_{i=1}^{n} n_{d^{x_i}(x_i)}(r) \\ &\leq n J_{\mathcal{G}(R,r)} \max_{d' \in D} \{n_{d'}(r)\}. \end{aligned} \tag{2.19}$$

By (2.18) and (2.19), we have

$$\frac{1}{n \max_{d' \in D} \{n_{d'}(r)\}} J_{\mathcal{N}(r)} \leq J_{\mathcal{G}(R,r)} \leq n J_{\mathcal{N}(r)} \max_{d' \in D} \{n_{d'}(r)\}. \tag{2.20}$$

Under the protocol, the number of α that $J_{\mathcal{N}(r)}$ contains is non-increasing at each remapping. When $J_{\mathcal{N}(r)}$ contains a fixed number of α, $J_{\mathcal{N}(r)}$ converges to a constant value under the controller in (2.6). It can be seen that $J_{\mathcal{G}(R,r)}$ is constant when $J_{\mathcal{N}(r)}$ is constant. At a fixed $J_{\mathcal{G}(R,r)}$, Situation 2 applies. In Situation 1, with a changing $J_{\mathcal{G}(R,r)}$ and due to the enlarging $\mathcal{M}_{x'}(t_l)$ and finite size of the objective map, the number of α that $J_{\mathcal{N}(r)}$ contains continues decreasing and converge to zero, and there are no two robots that are mapped to the same map robot. This means that the robot team X reaches a one-to-one mapping with the objective map D.

Situation 2: Assume that $\mathcal{G}(R, r)$ stays the same. According to the definition of $\mathcal{G}(R, r)$, a new mapping decision triggers a change in $\mathcal{G}(R, r)$. As the robot team is distributed, mapping decisions are made once at a time. Therefore, fixed $\mathcal{G}(R, r)$ means that no new mappings are made. On the assumption of fixed $\mathcal{G}(R, r)$, $J_{\mathcal{G}(R,r)}$ converges to zero under the controller in (2.6). With $J_{\mathcal{G}(R,r)} \to 0$, we have $J_{\mathcal{N}(r)} \to 0$ by (2.20). Thus, $J_{\mathcal{N}(r)}$ contains zero number of α, and there are no two robots that are mapped to the same map robot. This means that the robot team X has reached a one-to-one mapping with the objective map D. ■

Remark 2.3.4. *Note that for Protocol 2.3.1, each robot first is initiated by Initialisation. Then, the Conflict Locating process is triggered whenever Conflict Checking founds $W_{x_i} \neq \emptyset$ in (2.10). If the result of the Conflict Locating process is that $x'' \neq x_i$, then the Conflict Resolving process finds* *the new map robot $d^{x_i}(x_i)$ for robot x_i. Under the control law, a conflict, that there are robot x' and x'' mapped to the same map robot, is exposed to Conflict Checking, such robots x' and x'' become neighbours eventually. Then, as x' and x'' are remapped to different map robots through Conflict Resolving, the conflict is resolved. By enlarging both $\mathcal{M}_{x'}(t_l)$ and $\mathcal{M}_{x''}(t_l)$ to include the map robots of their neighbours, the objective map is reached after some iterations because of the finite team size. The proof of arrival at the objective map under Protocol 2.3.1 is given in Appendix 2.3.1.*

With the advantage of gradually enlarged $\mathcal{M}_{x_i}(t_l)$ and finite dimension of the objective map, all robots are mapped to a unique map robot. For robot x_i, it finds itself in a neighbourhood where $X'_{x_i} = G_{x_i}(R)$. Thus, the desired local optimum is found and robots cooperate to reach the current map formation. The key is that if two neighbouring robots are mapped to the same map robot, then one of local optima should be discarded and Protocol 2.3.1 is triggered to find a new local optimum. Although Protocol 2.3.1 is efficient in converging to the objective map, little consideration is taken for shortening the path to reach their final formation. As it is advantageous to shorten the path so that the robot team could work more efficiently, a novel protocol is proposed.

2.3.2 Path Shortening Protocol

In this section, a novel formation scheme is devised to achieve an improved performance in terms of shortening the path. Similar with the formation control setting in Protocol 2.3.1, given sensing range R, robot x_i's neighbours are grouped into $G_{x_i}(R)$. The current robot formation X are mapped to the current map formation $d(X)$, and the map robot sets $d(G_{x_i}(R))$ and the memory sets $\mathcal{M}_{x_i}(t_l)$ are defined in a similar fashion. The same restrictions on sensing and communication apply as those with Protocol 2.3.1 but additional definitions are needed in the new protocol.

Definition 2.3.2. *After Initialisation, if there exists some robot $x' \in G_{x_i}(R)$ such that is a neighbor of robot x_i and mapped to the same map robot as robot x_i by $d^{x'}(x') = d^{x_i}(x_i)$, then robot x_i's neighbourhood $G_{x_i}(R)$ and $d(G_{x_i}(R))$ is a conflict neighbourhood.*

In a non-conflict neighbourhood, two situations may exist: (i) the robot formation is *not yet stabilised* (there exists $x' \in X$ such that $u_{x'} \neq 0$) and is approaching a conflict neighbourhood; (ii) the robot formation is already *stabilised* ($u_{x'} = 0$ for any $x' \in X$) under a one-to-one mapping. In both cases, remapping is not needed. Otherwise, remapping is needed. Below, based on the map robots of robot x_i's control neighbourhood set $\mathcal{G}_{x_i}(R,r)$, robot x_i can have the private local map formation $d^{x_i}(\mathcal{G}_{x_i}(R,r))$ on its neighbouring robot group $\mathcal{G}_{x_i}(R,r)$ by,

$$d^{x_i}(\mathcal{G}_{x_i}(R,r)) = \arg \min_{d^{x_i}(\mathcal{G}_{x_i}(R,r)) \in D} J^d_{x_i}, \tag{2.21}$$

where

$$J^d_{x_i} = \sum_{x' \in \mathcal{G}_{x_i}(R,r)}^{x' \neq x_i} |d^{x'}(x') - d^{x_i}(x')|,$$

is a difference of assessments between robot x_i and its neighbours.

To assess the effect on the wondering paths for the robot team, we introduce a path measure in Definition 2.3.3.

Definition 2.3.3. *For robot x_i, the mapping misalignment between $\mathcal{G}_{x_i}(R,r)$ and $d^{x_i}(G_{x_i}(R))$ is defined as*

$$T_{x_i} = \sum_{x' \in \mathcal{G}_{x_i}(R,r)} |(x_i - x') - (d^{x_i}(x_i) - d^{x_i}(x'))|,$$

where the size of $G_{x_i}(R)$ is no smaller than that of $\mathcal{G}_{x_i}(R,r)$.

When the overall mapping misalignment $T = \sum_{i=1}^{n} T_{x_i} = 0$ occurs, the current robot formation X has converged to the current map formation $d(X)$. However, such map formation is not necessarily the objective map. Thus, to guarantee that the aforementioned current map formation is the objective map, Theorem 2.3.1 is proposed.

Theorem 2.3.1. *Under a connected formation graph, if $T_{x'} = 0$ for any $x' \in X$, then the mapping from the current robot formation to the current map formation is one-to-one.*

Proof of Theorem 2.3.1

Proof: Assume that i) the mapping from X to D is not one-to-one; ii) no robot is mapped to map robot $d_i \in D$; iii) map robot $d_j \in D$ is a neighbor of map robot d_i with $|d_i - d_j| \leq r$, and iv) d_j is mapped to x_j. As the formation is connected in three levels in Assumption 2.2.1 and $\mathcal{G}_{x_j}(R,r) \neq \emptyset$, by Definition 2.3.3, there exists robot $x_{j^1} \in \mathcal{G}_{x_j}(R,r)$ as robot x_j's neighbor, mapped to d_{j^1}, where $d_{j^1} \neq d_i$. As $|x_j - x_{j^1} - d_j + d_{j^1}| = 0$, we have

$x_j - x_{j^1} = d_j - d_{j^1}$. Similarly, there exists map robot d_{j^2} as map robot d_{j^1}'s neighbor, mapped to x_{j^2} and so on.

Repeat the process until either i) an isolated map robot d_{j^m} is found which no robot is mapped to, or ii) $d_{j^m} = d_j$ is found that establishes a circle. If an isolated map robot is found, then the process above is repeated for d_{j^m}. Subsequently, either another such isolated map robot or a circle is found. For cases where isolated map robots are found with no circles, robots are mapped to the same map robot d_j and this contradicts the assumption.

Assume that a circle is established and robot $x_{j'}$ is mapped to d_j. If $x_{j'}$ is a neighbor to x_j, then we have $x_{j'} = x_j$ due to $x_{j'} - x_j = d_j - d_j$. This contradicts the definition of a robot formation. If $x_{j'}$ is not a neighbor of x_j and $x_{j'} \neq x_j$, then we find two paths on the map that start from certain robot x_k and reach either x_j by x_k, x_{k_1}, ..., x_{k_p}, x_j or $x_{j'}$ by $x_{k'}$, $x_{k'_1}$, ..., $x_{k'_{p'}}$, $x_{j'}$. We can denote the map robot of x_{k_q} by d_{K_Q} and the map robot of $x_{k'_{q'}}$ by $d_{K'_{Q'}}$. According to Definition 2.3.3, we have

$$\begin{aligned}
x_k - x_j &= (x_k - x_{k_1}) + (x_{k_1} - x_{k_2}) + \cdots \\
&\quad + (x_{k_{p-1}} - x_{k_p}) + (x_{k_p} - x_j) \\
&= (d_K - d_{K_1}) + (d_{K_1} - d_{K_2}) + \cdots \\
&\quad + (d_{K_{P-1}} - d_{K_P}) + (d_{K_P} - d_j) \\
&= d_K - d_j \\
&= (d_K - d_{K'_1}) + (d_{K'_1} - d_{K'_2}) + \cdots \\
&\quad + (d_{K'_{P'-1}} - d_{K'_{P'}}) + (d_{K'_{P'}} - d_j) \\
&= (x_K - x_{K'_1}) + (x_{K'_1} - x_{K'_2}) + \cdots \\
&\quad + (x_{K'_{P'-1}} - x_{K'_{P'}}) + (x_{K'_{P'}} - x_j) = x_k - x_{j'}.
\end{aligned}$$

Thus, $x_j = x_{j'}$ and this contradicts the definition of a robot formation. This completes the proof. ■

We now develop the component of Protocol 2.3.2 that functions to reduce the difference of assessments and shorten the wandering path, in the context of making a new mapping decision in a conflict neighbourhood. To achieve this, whenever robot x_i reaches its conflict neighbourhood, it is remapped to another map robot that is: (i) not mapped to any of its neighbours, (ii) does not belong to $\mathcal{M}_{x_i}(t_l^-)$, and (iii) contains most map robot neighbours that its neighbours are mapped to. If there exists a group of such map robots, then robot x_i is remapped to the most suitable map robot.

To proceed, some new terms need to be defined. Based on D and $d(G_{x_i}(R))$, a set of $\mathcal{K}_{G_{x_i}(R)}(t_l^-)$ for robot x_i is defined below.

$$\mathcal{K}_{G_{x_i}(R)}(t_l^-) = \left\{ \begin{array}{ccc} d_1^{\mathcal{K}_{G_{x_i}(R)}(t_l^-)} & d_2^{\mathcal{K}_{G_{x_i}(R)}(t_l^-)} & \dots \end{array} \right\}, \tag{2.22}$$

where for any $d' \in D$, $d' \in \mathcal{K}_{G_{x_i}(R)}(t_l^-)$ if we have

i) $d' \notin d(G_{x_i}(R))$ or $d' = d^{x_i}(x_i)$, and

ii) $d' \notin \mathcal{M}_{x_i}(t_l^-)$ and $\mathcal{H}_{d'}\left(d(G_{x_i}(R))\right)$ has no fewer points than $\mathcal{H}_{d''}(d(G_{x_i}(R)))$ for any $d'' \in D$.

For a set D' of points under R, $\mathcal{H}_{d_i}(D')$ is defined below.

$$\mathcal{H}_{d_i}(D') = \left\{ \begin{array}{ccc} d_1^{\mathcal{H}_{d_i}(D')} & d_2^{\mathcal{H}_{d_i}(D')} & \dots \end{array} \right\},$$

where for any $d' \in D'$, $d' \in \mathcal{H}_{d_i}(D')$ if $|d' - d_i| \leq R$. Based on $\mathcal{K}_{G_{x_i}(R)}(t_l^-)$, Protocol 2.3.2 is given below.

Protocol 2.3.2. *For any robot x_i, whenever robots x_i has a conflict neighbourhood, it is remapped to a new map robot by (2.23).*

$$\begin{array}{l} For\ \ any\ \ x' \in \mathcal{G}_{x_i}(R,r), \\ \quad i)\ \ d^{x_i}(x') = d^{x'}(x)'\ \ if\ \ x' \neq x_i,\ \ and \\ \quad ii)\ \ d^{x_i}(x') = \arg \min\limits_{d^{x_i}(x') \in \mathcal{K}_{G_{x_i}(R)}(t_l^-)} T_{x'}\ \ if\ \ x' = x_i. \end{array} \tag{2.23}$$

The identity of the new map robot $d^{x_i}(x_i)$ in (2.23) is sent to the robots in $G_{x_i}(R)$ through the communication channel. After a new mapping decision is made, $\mathcal{M}_{x_i}(t_l)$ is updated by

$$\mathcal{M}_{x_i}(t_l) = \mathcal{M}_{x_i}(t_l^-) \oplus d(G_{x_i}(R)).$$

Proof of Arrival at a Desired Formation under Protocol 2.3.2

Proof: Similar to the proof of Protocol 2.3.1, two different situations are considered.

Situation 1: Assume that the map formation is changing. By Definition 2.3.3, we have

$$T_{x_i} = \sum_{x_j \in \mathcal{G}_{x_i}(R,r)} \left| (x_i - d^{x_i}(x_i)) - (x_j - d^{x_i}(x_j)) \right|, \tag{2.24}$$

and

$$\begin{aligned}
T_{x_i} &= \sum_{x_j \in \mathcal{G}_{x_i}(R,r)} |(x_i - d^{x_i}(x_i)) - (x_j - d^{x_j}(x_j)) - (d^{x_j}(x_j) - d^{x_i}(x_j))| \\
&\leq \sum_{x_j \in \mathcal{G}_{x_i}(R,r)} |(x_i - d^{x_i}(x_i)) - (x_j - d^{x_j}(x_j)| \\
&\quad + \sum_{x_j \in \mathcal{G}_{x_i}(R,r)} |d^{x_j}(x_j) - d^{x_i}(x_j)| \\
&\leq \sum_{x_j \in \mathcal{G}_{x_i}(R,r)} |(x_i - d^{x_i}(x_i)) - (x_j - d^{x_j}(x_j)| \\
&\quad + \sum_{x_j \in \mathcal{G}_{x_i}(\infty,r)} |d^{x_j}(x_j) - d^{x_i}(x_j)|.
\end{aligned}$$

In the controller (2.6), robot x_i is always trying to find neighbours $\mathcal{G}_{x_i}(R, r)$ that are mapped to $d^{x_i}(x_i)$'s map neighbours. In a most-neighbor mapping scheme, the value of $|d^{x_j}(x_j) - d^{x_i}(x_j)|$ exists only when robot x_i reaches its conflict neighbourhood. Then, given the remapping by Protocol 2.3.2, $\sum_{x_j \in \mathcal{G}_{x_i}(\infty,r)} |d^{x_j}(x_j) - d^{x_i}(x_j)|$ is non-increasing.

Since the robot formation has finite dimensions, each robot can eventually find a neighbourhood where it no longer needs to be remapped. It is reasonable to assume that after some finite steps of iteration, $\sum_{x_j \in \mathcal{G}_{x_i}(\infty,r)} |d^{x_j}(x_j) - d^{x_i}(x_j)| \to 0$. Ultimately, $\sum_{x_j \in \mathcal{G}_{x_i}(\infty,r)} |d^{x_j}(x_j) - d^{x_i}(x_j)|$ is 0 for all $i = 1, \ldots, n$.

Situation 2: Assume that the map formation is fixed. This is only possible when either no robot has arrived its conflict neighbourhood or the map formation is under a one-to-one mapping. We define that

$$\begin{aligned}
T_{\mathcal{G}(R,r)} &= \sum_{i=1}^{n} \sum_{x_j \in \mathcal{G}_{x_i}(R,r)} | - x_i + x_j + d^{x_i}(x_i) - d^{x_j}(x_j)|, \\
&\geq \sum_{i=1}^{n} T_{x_i} - \sum_{i=1}^{n} \sum_{x_j \in \mathcal{G}_{x_i}(\infty,r)} |d^{x_j}(x_j) - d^{x_i}(x_j)|. \qquad (2.25)
\end{aligned}$$

As the robot formation is connected and the map formation is fixed, $T_{\mathcal{G}(R,r)}$ converges to zero under the controller in (2.6). Moreover, by Protocol 2.3.2,

$\sum_{i=1}^{n} \sum_{x_j \in \mathcal{G}_{x_i}(\infty, r)} |d^{x_j}(x_j) - d^{x_i}(x_j)|$ converges to zero. Thus, by (2.25), we have $T_{x_i} \to 0$, and we have

$$\sum_{i=1}^{n} \sum_{x_j \in \mathcal{G}_{x_i}(R,r)} |(x_i - d^{x_i}(x_i)) - (x_j - d^{x_i}(x_j))| = 0.$$

Therefore, by Definition 2.3.3 and Theorem 2.3.1, the mapping from the robot team X to the map formation $d(X)$ is one-to-one. Under the controller in (2.6), X is driven to a desired formation. ■

Remark 2.3.5. *For robot x_i, $d^{x_i}(x_i)$ and $d^{x'}(x')$ for $x' \in G_{x_i}(R)$ are possibly misleading for being suitable only for the current moment. A new mapping decision is generated, if robot x_i reaches its conflict neighbourhood and the old ones lose its optimality in (2.23). For robot x_i in its current neighbourhood, if no neighbor is mapped to the same map robot as robot x_i, then this neighbourhood is a non-conflict one. In a non-conflict neighbourhood, the current mapping decision remains suitable, and a new one is not needed.*

Remark 2.3.6. *Note that it is computationally costly to find the solution of (2.21) and such solution is not capable of avoiding conflict neighbourhoods. Therefore, to derive an approximate solution for (2.21) that is both efficient and permissible, a most-neighbor scheme is adopted in Protocol 2.3.2 by introducing $\mathcal{K}_{G_{x_i}(R)}(t_l^-)$ from (2.22) into (2.23). As a smaller mapping misalignment is preferred in (2.23), reduced wandering paths are observed in Section 2.4. The proof of arrival at the objective map under protocol 2.3.2 is given in Appendix 2.3.2. Many studies have been done on second order multi-agent systems [5, 15, 16, 18, 25, 26, 27, 29], to which Protocol 2.3.1 and Protocol 2.3.2 in this chapter can readily be adapted.*

Remark 2.3.7. *The memory set $\mathcal{M}_{x_i}(t_l)$ is used here to exclude the possibility of any dynamic infinite loop. For example, if robot x_i is mapped to map $d^{x_i}(x_i(t_0))$, then after a while, it is remapped to $d^{x_i}(x_i(t_1))$. Without $\mathcal{M}_{x_i}(t_l)$, it can be remapped to $d^{x_i}(x_i(t_l)) = d^{x_i}(x_i(t_0))$ again. Alternatively, when the size of the robot formation is sufficiently big, it can be chronologically mapped to*

$$\begin{aligned} &d^{x_i}(x_i(t_0)) \to d^{x_i}(x_i(t_1)) \to d^{x_i}(x_i(t_2)) \to \cdots \\ &\to d^{x_i}(x_i(t_{l-1})) \to d^{x_i}(x_i(t_l)), \end{aligned}$$

where $d^{x_i}(x_i(t_l)) = d^{x_i}(x_i(t_0))$. Thus, an infinite loop is established. With $\mathcal{M}_{x_i}(t_l)$ and under Protocol 2.3.2 (similarly for Protocol 2.3.1), this situation can be avoided.

2.3.3 Maintaining Formation Connectivity

In the previous sections, two different protocols based on the local optima are designed for the controller in (2.6) to drive a group of robots such that the robot team can interact and reach a globally desired formation. However, the successful transformation relies on the connectivity of the formation. In this section, we propose a condition to guarantee such connectivity.

To guarantee the formation connectivity, we derive an inequality condition on R and r (in Theorem 2.3.3). Based on X and D, we can obtain

$$\begin{aligned}\delta X &= \left\{ \begin{array}{ccc} \delta x_1 & \delta x_2 & \dots \end{array} \right\}, \; \delta \mathbf{X}^T = \left[\begin{array}{ccc} \delta x_1^T & \delta x_2^T & \dots \end{array} \right], \\ \delta D &= \left\{ \begin{array}{ccc} \delta d_1 & \delta d_2 & \dots \end{array} \right\}, \; \delta \mathbf{D}^T = \left[\begin{array}{ccc} \delta d_1^T & \delta d_2^T & \dots \end{array} \right],\end{aligned}$$

where $x' - x'' \in \delta X$ for any $x', x'' \in X$ and $d' - d'' \in \delta D$ for any $d', d'' \in D$. After the robots have been provided with the map formation $d(G_{x_i}(R))$ by communication and either Protocol 2.3.1 or Protocol 2.3.2, the connection matrix C between $\delta \mathbf{X}$ and $\delta \mathbf{D}$ can be obtained below.

$$C = \begin{bmatrix} c_{11} & c_{12} & c_{13} & \cdots \\ c_{21} & c_{22} & c_{23} & \cdots \\ c_{31} & c_{32} & c_{33} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix}, \tag{2.26}$$

where

$$c_{ab} = \begin{cases} I, & \begin{array}{c} \text{where } d' = d^{x'}(x'), d'' = d^{x''}(x''), x'' \in \mathcal{G}_{x'}(\infty, r) \\ \text{for } \delta x_a = x' - x'' \text{ and } \delta d_b = d' - d''; \end{array} \\ 0, & \qquad\qquad \text{otherwise.} \end{cases}$$

It can be seen that the connection matrix C is dependent on the mapping, from the minimisation in (2.9), (2.16) and (2.23) by Protocol 2.3.1 or Protocol 2.3.2. In (2.26), R is sufficiently large such that $G_{x'}(R) = X$ covers the whole team for any $x' \in X$, and the mapping can be regarded as global. Then, $\delta \mathbf{X}$ and $C^* \delta \mathbf{D}$ have a one-to-one mapping with global optimum below.

$$C^* = \arg \min_C S, \tag{2.27}$$

where the mapping energy is

$$S = (\delta \mathbf{X} - C \delta \mathbf{D})^T (\delta \mathbf{X} - C \delta \mathbf{D}).$$

If R is small, the mapping can only be regarded as locally optimal. Then, we have

$$\delta X_i = \left\{ \begin{array}{ccc} \delta x_1^i & \delta x_2^i & \dots \end{array} \right\}, \; \delta \mathbf{X}_i^T = \left[\begin{array}{ccc} e_1^i \delta x_1^T & e_2^i \delta x_2^T & \dots \end{array} \right],$$

where for any $x', x'' \in X$, $x' - x'' \in \delta X_i$ if $x', x'' \in \mathcal{G}_{x_i}(R, r)$; for $\delta x_k = x' - x''$, $e^i_k = 0$ if $x' \notin \mathcal{G}_{x_i}(R, r)$ or $x'' \notin \mathcal{G}_{x_i}(R, r)$, and $e^i_k = I$ if $x', x'' \in \mathcal{G}_{x_i}(R, r)$. The connection matrix C_i between $\delta \mathbf{X}_i$ and $\delta \mathbf{D}$ can be obtained below.

$$C_i = \begin{bmatrix} c^i_{11} & c^i_{12} & c^i_{13} & \cdots \\ c^i_{21} & c^i_{22} & c^i_{23} & \cdots \\ c^i_{31} & c^i_{32} & c^i_{33} & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix},$$

where

$$c^i_{ab} = \begin{cases} I, & \begin{array}{c} \text{where } d' = d^{x'}(x'), d'' = d^{x''}(x''), x'' \in \mathcal{G}_{x'}(R, r) \\ \text{for } \delta x^i_a = x' - x'' \text{ and } \delta d_b = d' - d''; \end{array} \\ 0, & \qquad\qquad \textit{otherwise.} \end{cases}$$

Thus, $\delta \mathbf{X}_i$ and $C^*_i \delta \mathbf{D}$ have a mapping with the local optimum where

$$C^*_i = \arg \min_{C_i} S_i. \tag{2.28}$$

where the mapping energy is

$$S_i = (\delta \mathbf{X}_i - C_i \delta \mathbf{D})^T (\delta \mathbf{X}_i - C_i \delta \mathbf{D}).$$

It can be seen that

$$\delta X = \delta X_1 \oplus \delta X_2 \oplus \cdots \oplus \delta X_n, \text{ and}$$

there exists r such that we can have

$$\delta X_1 \odot \delta X_2 \odot \cdots \odot \delta X_n = \emptyset.$$

Remark 2.3.8. *Note that the global optimum and the local optima are generated under the same set of optimisation equations from either Protocol 2.3.1 or Protocol 2.3.2. Although the global optimum is globally optimal, it is probably less desirable than the local optima from the mapping energy's point of view. Thus, the global optimum is naturally an upper bound for all the local optima. Below, the global optimum and the local optima are used to introduce a connectivity maintaining condition on the sensing range and control range.*

We are now ready to present the connectivity conditions for robot formation.

Theorem 2.3.2. *For mapping energies S and S_i in (2.27) and (2.28), we have*

$$S \geq S_1 + S_2 + \cdots + S_n.$$

Proof of Theorem 2.3.2

Proof: By $\delta X = \delta X_1 \oplus \delta X_2 \oplus \cdots \oplus \delta X_n$, $\delta X_1 \odot \delta X_2 \odot \cdots \odot \delta X_n = \emptyset$, we can have

$$\begin{aligned} \delta\mathbf{X} &= \delta\mathbf{X}_1 + \delta\mathbf{X}_2 + \cdots + \delta\mathbf{X}_n, \\ \delta\mathbf{X}^T\delta\mathbf{X} &= \delta\mathbf{X}_1^T\delta\mathbf{X}_1 + \delta\mathbf{X}_2^T\delta\mathbf{X}_2 + \cdots + \delta\mathbf{X}_n^T\delta\mathbf{X}_n. \end{aligned}$$

there exists C_i^* that yields a local optimum in (2.28), and we can have

$$\begin{aligned} (\delta\mathbf{X}_i - C_i^*\delta\mathbf{D}_i)^T(\delta\mathbf{X}_i - C_i^*\delta\mathbf{D}_i) \;-\; (\delta\mathbf{X}_i - C^*\delta\mathbf{D}_i)^T(\delta\mathbf{X}_i - C^*\delta\mathbf{D}_i) \leq 0, \\ -\delta\mathbf{X}_i^T C_i^*\delta\mathbf{D}_i + \delta\mathbf{D}_i^T C_i^{*T} C_i^*\delta\mathbf{D}_i \;+\; \delta\mathbf{X}_i^T C^*\delta\mathbf{D}_i - \delta\mathbf{D}_i^T C^{*T} C^*\delta\mathbf{D}_i \leq 0, \\ \delta\mathbf{D}_i^T C^{*T} C^*\delta\mathbf{D}_i - \delta\mathbf{X}_i^T C^*\delta\mathbf{D}_i \geq \delta\mathbf{D}_i^T C_i^{*T} C_i^*\delta\mathbf{D}_i - \delta\mathbf{X}_i^T C_i^*\delta\mathbf{D}_i. \end{aligned}$$

Then, it follows that

$$\begin{aligned} & (\delta\mathbf{X} - C^*\delta\mathbf{D})^T(\delta\mathbf{X} - C^*\delta\mathbf{D}) \\ = \; & \delta\mathbf{X}^T\delta\mathbf{X} - 2\delta\mathbf{X}^T C^*\delta\mathbf{D} + \delta\mathbf{D}^T C^{*T} C^*\delta\mathbf{D} \\ = \; & \delta\mathbf{X}_1^T\delta\mathbf{X}_1 + \delta\mathbf{X}_2^T\delta\mathbf{X}_2 + \cdots + \delta\mathbf{X}_n^T\delta\mathbf{X}_n \\ & +\delta\mathbf{D}^T(C_1^{*T}C_1^* + C_2^{*T}C_2^* + \cdots + C_n^{*T}C_n^*)\delta\mathbf{D} \\ & -2(\delta\mathbf{X}_1 + \delta\mathbf{X}_2 + \cdots + \delta\mathbf{X}_n)^T C^*\delta\mathbf{D} \\ \geq \; & \delta\mathbf{X}_1^T\delta\mathbf{X}_1 + \delta\mathbf{X}_2^T\delta\mathbf{X}_2 + \cdots + \delta\mathbf{X}_n^T\delta\mathbf{X}_n \\ & +\delta\mathbf{D}^T(C_1^{*T}C_1^* + C_2^{*T}C_2^* + \cdots + C_n^{*T}C_n^*)\delta\mathbf{D} \\ & -2\delta\mathbf{X}_1^T C_1^*\delta\mathbf{D} + 2\delta\mathbf{X}_2^T C_2^*\delta\mathbf{D} + \cdots + 2\delta\mathbf{X}_n^T C_n^*\delta\mathbf{D} \\ = \; & (\delta\mathbf{X}_1 - C_1^*\delta\mathbf{D}_1)^T(\delta\mathbf{X}_1 - C_1^*\delta\mathbf{D}_1) + (\delta\mathbf{X}_2 - C_2^*\delta\mathbf{D}_2)^T(\delta\mathbf{X}_2 - C_2^*\delta\mathbf{D}_2) \\ & + \cdots + (\delta\mathbf{X}_n - C_n^*\delta\mathbf{D}_n)^T(\delta\mathbf{X}_n - C_n^*\delta\mathbf{D}_n). \end{aligned}$$

Thus, we have

$$\begin{aligned} (\delta\mathbf{X} - C^*\delta\mathbf{D})^T(\delta\mathbf{X} - C^*\delta\mathbf{D}) \geq \; & (\delta\mathbf{X}_1 - C_1^*\delta\mathbf{D}_1)^T(\delta\mathbf{X}_1 - C_1^*\delta\mathbf{D}_1) \\ & +(\delta\mathbf{X}_2 - C_2^*\delta\mathbf{D}_2)^T(\delta\mathbf{X}_2 - C_2^*\delta\mathbf{D}_2) \\ & + \cdots + (\delta\mathbf{X}_n - C_n^*\delta\mathbf{D}_n)^T(\delta\mathbf{X}_n - C_n^*\delta\mathbf{D}_n)) \, , \end{aligned}$$

which completes the proof. ■

Theorem 2.3.3. *Under Assumption 2.2.1, if* $R \geq \sqrt{\frac{S}{2}} + r$ *and each robot behaves according to Initialisation and either Protocol 2.3.1 or Protocol 2.3.2, then the group connectivity is maintained.*

Proof: Refer to Appendix 2.3.3. ■

Proof of Theorem 2.3.3

Proof: It can be seen that for any $x_j \in \mathcal{G}_{x_i}(R, r)$, it follows that

$$\frac{1}{2}S \geq \frac{1}{2}(S_1 + S_2 + \cdots + S_n) \geq (x_i - x_j - d^{x_i}(x_i) + d^{x_j}(x_j))^2.$$

Then, we have

$$\begin{aligned}\sqrt{\frac{S}{2}} &\geq |x_i - x_j \ - d^{x_i}(x_i) + d^{x_j}(x_j)| \\ &\geq |x_i - x_j| - |d^{x_i}(x_i) - d^{x_j}(x_j)|, \\ \sqrt{\frac{S}{2}} + \max_{x_j \in \mathcal{G}_{x_j}(R,r)} |d^{x_i}(x_i) \ &- d^{x_j}(x_j)| \geq \max_{x_k \in \mathcal{G}_{x_j}(R,r)} |x_k - x_l|.\end{aligned}$$

Thus, to maintain the link between robot x_i and robot x_j, we have

$$\begin{aligned}R &\geq \sqrt{\frac{S}{2}} + \max_{x_j \in \mathcal{G}_{x_j}(R,r)} |d^{x_i}(x_i) - d^{x_j}(x_j)|, \\ &\geq \sqrt{\frac{S}{2}} + r.\end{aligned}$$

Thus, given the control range r, if $R \geq \sqrt{\frac{S}{2}} + r$, then the formation connectivity is maintained. Subsequently, when the initial robot formation is connected by Assumption 2.2.1, it always remains connected under a related protocol.

■

Remark 2.3.9. *Note that by Assumption 2.2.1 and Protocol 2.3.1 or 2.3.2, the connectivity of the formation is maintained. Under such connectivity, either Protocol 2.3.1 or Protocol 2.3.2 is able to drive local optima to a global optimum. Moreover, Assumption 2.2.1 states three levels of connectivity, and Theorem 2.3.3 provides a condition on the connectivity in the three levels.*

Remark 2.3.10. *Note that maintaining the formation connectivity and map formation in a time varying multi-robot system is challenging. We claim that initial robot and map formations have an important influence. Through extensive simulations, we observe that if the topologies of the initial robot formation is connected under the three link types in Section 3.2, then it is highly likely that the topologies remain connected. Also, with Assumption 2.2.1 and under the maintained connected topology, the protocols presented in Section 2.3.1 and Section 2.3.2 can always find their solutions.*

2.4 Simulation Examples

In this section, we conduct a set of simulations to verify the performance of the distributed formation control scheme. The sensing range is $R = 20$ meters and the control range is $r = 18$ meters. The displacements within the sensing range are used to make a mapping decision, and then displacements within the control range are used to produce the control forces. The objective is twofold: 1) to reach a one-to-one mapping from all the robots to all the predefined points in a distributed manner, and 2) to drive all the robots to a desired formation from the objective map. The objective map is $D = \left\{ d_1 \quad d_2 \quad \ldots \quad d_9 \right\}$, where

$$\begin{aligned} d_1 &= \begin{bmatrix} -13.5 \\ 29.4 \end{bmatrix}, \; d_2 = \begin{bmatrix} 2.9 \\ 24.9 \end{bmatrix}, \; d_3 = \begin{bmatrix} -8.7 \\ 13.1 \end{bmatrix}, \\ d_4 &= \begin{bmatrix} 3.1 \\ 11.4 \end{bmatrix}, \; d_5 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \; d_6 = \begin{bmatrix} -10.3 \\ -11.2 \end{bmatrix}, \\ d_7 &= \begin{bmatrix} 15 \\ 0 \end{bmatrix}, \; d_8 = \begin{bmatrix} 21.8 \\ -10.7 \end{bmatrix}, \; d_9 = \begin{bmatrix} 29.1 \\ 2.2 \end{bmatrix}. \end{aligned}$$

With different initial positions for the nine robots, a number of simulation runs were conducted. In one of the simulations, Figure 2.2 and Figure 2.3 show the snapshots of the 2-D path and the objective map, respectively. For the legend $*(\times)$ used in Figure 2.2, $*$ denotes a number that is the identity of a robot, and $\times$ denotes a number that is the identity of robot $*$'s map robot. It can be seen that ultimately, the robot team reaches a desired formation from the objective map. Figure 2.4 shows that degree of the set $d(X) = \left\{ d^{x_1}(x_1) \quad d^{x_2}(x_2) \quad \ldots \quad d^{x_9}(x_9) \right\}$ evolves in a nondecreasing way in time and reaches its maximum number, as is proved for Protocol 2.3.2. The same was found in the simulations with Protocol 2.3.1.

It can be seen from Table 2.1 that when the initial positions of the robots are changed, the robots reach different desired formations under different map formations by Protocol 2.3.1. For example, in the second row at third column of Table 2.1, it can be seen from $\{d_6, d_7, d_5, d_9, d_4, d_7, d_1, d_3, d_2\}$ that $d^{x_1}(x_1) = d_6$, $d^{x_2}(x_2) = d_7$, $d^{x_3}(x_3) = d_5$, $d^{x_4}(x_4) = d_9$, $d^{x_5}(x_5) = d_4$, $d^{x_6}(x_6) = d_6$, $d^{x_7}(x_7) = d_1$, $d^{x_8}(x_8) = d_3$, and $d^{x_9}(x_9) = d_2$. Moreover, t_m is the time elapsed when the degree of $d(X)$ reaches 9, and t_s is the time when the simulation ceases with the mean square root of the control force smaller than 10. For the final robot formation $X(t_s) = \left\{ x_1 \quad x_2 \quad \ldots \quad x_9 \right\}$, the column under $d(X(t_s))$ shows that its final map formation is $d(X(t_s)) =$

$\{\ d^{x_1}(x_1)\ \ d^{x_2}(x_2)\ \ \dots\ \ d^{x_9}(x_9)\ \}$. Table 2.2 shows the results under Protocol 2.3.2.

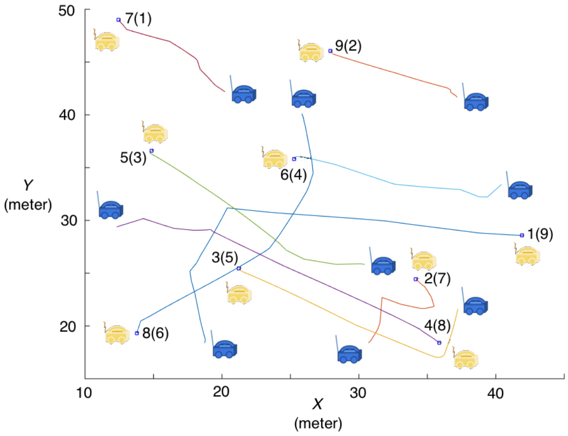


Figure 2.2: The paths under Protocol 2.3.1; light and dark cars are the initial and final positions.

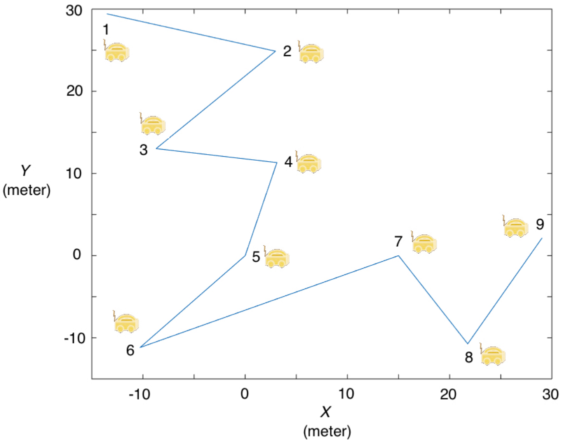


Figure 2.3: Snapshot of the objective map.

Comparing Table 2.1 with Table 2.2, we have the following observations. (i) Some robots are more likely to be mapped to some map robots than others. This implies that the final map formation is dependent on initial conditions, as discussed in Remark 2.3.10, which suggests the ability to self-organise. (ii) Protocol 2.3.1 is, on average, faster in reaching the full map formation. (iii) It takes roughly the same amount of wandering time for both protocols to reach the objective map. It is because Protocol 2.3.1 takes less consideration in shortening the path. In addition, we observe from all the simulations that the paths found from using Protocol 2.3.2 are on average shorter than those from using Protocol 2.3.1.

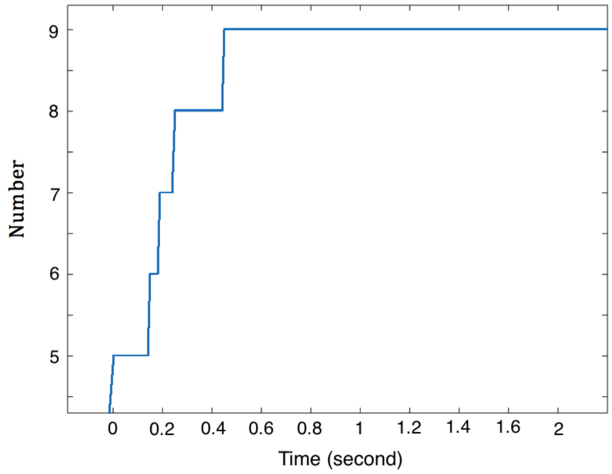


Figure 2.4: Degree of $d(X)$ under Protocol 2.3.2.

Table 2.1: Final mapping by Protocol 2.3.1.

t_m	t_s	$d(X(t_s))$
0.63	6.28	$\{d_6, d_7, d_5, d_9, d_4, d_6, d_1, d_3, d_2\}$
0.89	6.11	$\{d_9, d_7, d_5, d_6, d_3, d_4, d_8, d_1, d_2\}$
1.00	6.12	$\{d_9, d_7, d_5, d_8, d_3, d_4, d_6, d_1, d_2\}$
0.86	6.68	$\{d_6, d_9, d_5, d_3, d_4, d_7, d_1, d_2, d_8\}$
0.76	6.46	$\{d_9, d_7, d_3, d_5, d_4, d_8, d_6, d_1, d_2\}$
0.83	6.48	$\{d_9, d_7, d_4, d_5, d_8, d_6, d_1, d_3, d_2\}$
0.20	5.74	$\{d_9, d_5, d_4, d_7, d_8, d_3, d_6, d_1, d_2\}$
0.61	6.16	$\{d_6, d_9, d_7, d_3, d_5, d_4, d_8, d_1, d_2\}$
0.92	6.58	$\{d_9, d_6, d_7, d_5, d_3, d_1, d_8, d_4, d_2\}$

Table 2.2: Final mapping by Protocol 2.3.2.

t_m	t_s	$d(X(t_s))$
0.95	6.91	$\{d_6, d_9, d_7, d_5, d_4, d_1, d_2, d_3, d_8\}$
1.19	6.62	$\{d_6, d_7, d_9, d_5, d_4, d_1, d_8, d_2, d_3\}$
0.81	7.03	$\{d_7, d_5, d_9, d_8, d_1, d_6, d_4, d_2, d_3\}$
0.87	6.61	$\{d_6, d_5, d_7, d_1, d_4, d_9, d_2, d_3, d_8\}$
0.47	6.87	$\{d_4, d_3, d_7, d_9, d_5, d_6, d_8, d_2, d_1\}$
1.66	7.93	$\{d_9, d_3, d_1, d_5, d_4, d_2, d_6, d_8, d_7\}$
1.21	7.10	$\{d_6, d_5, d_9, d_8, d_1, d_7, d_2, d_3, d_4\}$
0.88	6.41	$\{d_9, d_5, d_1, d_7, d_4, d_8, d_2, d_3, d_6\}$
1.44	7.07	$\{d_6, d_9, d_7, d_8, d_5, d_3, d_2, d_1, d_4\}$

2.5 Summary

In this chapter, the formation control for a team of robots is studied. An initialisation protocol is designed to find the initial mapping through the local optima. Based on these initial local optima, two protocols are developed to reach the global optimum. In addition to the limited sensing range, a limited control range is proposed. Based on the control range and the relationship between the global optimum and local optima, a condition on the sensing range is given to guarantee the formation connectivity.

The theoretical deduction for system analysis may be complex but since the formation is distributed, the individual task for each robot is simple, which can be seen in our program codes for the protocols. Even though proofs of guaranteed arrival at the objective map are derived, the condition of connected topology could be further improved to reduce its conservativeness. It is quite possible that the objective map could still be reached even if robots suffer from an occasionally disconnected topology. Therefore, future work opens for conditions of mixed topologies for reaching the objective map.

Chapter 3

Scalable Formation with No Communication

In situations where robots need to keep electromagnetic silent in a formation, communication channels become unavailable. Moreover, as passive displacement sensors are used, limited sensing ranges are inevitable due to power insufficiency and limited noise reduction. To address the formation control problem for a scalable team of robots subject to the above restrictions, a flexible strategy is necessary. In this chapter, under the assumption that the data transmission among the robots is not available, a novel controller and a protocol are designed that do not rely on communication. As the controller only drives the robots to a partially desired formation, a distributed coordination protocol is proposed to resolve the imperfections. It is shown that the effectiveness of the controller and the protocol rely on the formation connectivity, and a condition is given on the sensing range. Simulations are conducted to illustrate the feasibility and advantages of the new design scheme developed.

3.1 Introduction

In this chapter, we investigate the formation control problem for a team of robots. The robots are equipped with displacement sensors of limited ranges, similar to [8, 10, 47], and the in-range robots are indistinguishable to the sensors. This means that no robot can identify its neighbouring robots, compared with [4]. Measurements are the displacements among in-range robots, which consist of the distances and the directions. Data communication among them is not available, which represents a more difficult problem than that in [8, 37, 51, 59, 61]. As a result, the robots can only observe the robot team from a local field of view. They have no fixed roles or positions, and are able to locally find the most suitable positions in the predefined reference formation, unlike [8, 10, 32, 37, 39, 52, 57, 59].

Our contribution in this chapter is that under the aforementioned restrictions, the robots are programmed to have the same behaviour pattern (control law) and the same thinking pattern (coordination protocol), thus, no extra effort is needed for system scalability. To make the desired robot formation scalable, a class of extended formations based on relative displacement framework are defined as the reference. The team size is scalable so that new robots can be connected and included to the *extension positions*, and existing ones can be disconnected and excluded. Since the desired robot formation is simplified by the displacement framework, the computation load is considerably reduced. In the scalable formation, robots need to find their most suitable positions through local mappings, instead of accessing global information as in [4, 6, 51, 58, 61, 62].

The suitable positions for different robots sometimes could be conflicting. To deal with this issue, a novel controller is designed rather than the popular control laws based on negative gradient and potential field in [35]. Each robot is driven by the control force such that a team of such robots interact locally and reach a partially desired formation with imperfect mappings. On the premise that the predefined desired formation is based on relative displacement, the imperfect mappings that lead to potential collisions can be exposed. Then, a shape matching based coordination protocol is devised to coordinate the robots and resolve the conflicts.

Notation. The notation used is standard. The superscript "T" stands for matrix transposition; $P \leq 0$ means that P is non-positive definite; $|\cdot|$ denotes the length of a vector. The bold and the non-bold styles of a symbol denote different variables; the italic and the non-italic styles of a symbol denote the same variable. Matrices are assumed to be compatible for algebraic operations.

3.2 Robot Formation and Displacement-Based Reference Formation

In the formation control problem that we address, robots only have displacement sensors of limited ranges, and the sensors cannot distinguish the robots. There are no communication channels among the robots and thus, the sensor measurements are the only real-time reference for the robots to move. The robots are equally important. They are independent and do not have fixed roles of leaders or followers or fixed positions in the objective formation. Through the sensor measurements, it is expected that the robots could interact and cooperate to reach a shape from the objective formation.

In our proposed solution, as the robots have no fixed positions in the objective formation, they need to make real-time mapping decisions on their current suitable mapping positions in the objective formation. As no data communication is available, the robots can only access sensor measurements and have to keep that information to their own. Even though the sensing ranges are limited, the in-range areas of different robots are intersected and accessible to the in-range robots. Thus, the intersected areas can be used by the in-range robots to make real-time mapping decisions. Moreover, as the sensor measurements are divided into intersected sets, the points in the objective formation can also be divided into different reference groups. Based on the reference groups from the objective formation and the intersected in-range areas from the sensors, the robots can make some mapping decisions and generate the control force.

3.2.1 Derivation of Robot Formation

Our formation control objective is to design the control law and the coordination protocol for each robot such that through local interaction, all the robots

$$x = \left\{ \begin{array}{cccc} x_1 & x_2 & \dots & x_n \end{array} \right\}, \tag{3.1}$$

are driven to the displacements from certain objective formation

$$F = \left\{ \begin{array}{ccccc} \dots & d_{-1} & d_0 & d_1 & \dots \end{array} \right\}, \tag{3.2}$$

where $x' \in \mathbf{R}^2$ for any $x' \in x$, and $d' \in \mathbf{R}^2$ for any $d' \in F$. This means that for any x' and $x'' \in x$, if there exist d' and $d'' \in F$ such that $x' - x'' = d' - d''$, then the robot team has reached a desired formation. The objective formation F is user-defined and predefined as the reference for the coordination protocol to produce mapping decisions, with which the control forces are generated.

It is assumed that the robots are not communicating with each other, and that the individual sensing range is limited. It means that robots have no knowledge of other robots outside their sensing ranges. The controller of each robot is provided with only the displacements to its neighbours and the sensor cannot distinguish its neighbours. Their displacement sensors only detect those that are inside the sensing ranges, and the robots can only keep the measurements to themselves. Let R denote the sensing range. For any x' and $x'' \in x$ if $|x' - x''| \leq R$, then there is a link between robot x' and x''; all the links constitute the robot formation topology. If there is a path of links between any two robots, then we say that the robot formation is *connected.*

For robot $x_i \in x$ and under the sensing range R, its group g_i of neighbours is defined below.

$$g_i = \left\{ \begin{array}{ccc} x_1^{g_i} & x_2^{g_i} & \dots \end{array} \right\}, \tag{3.3}$$

where for any $x' \in x$, if $|x' - x_i| \leq R$, then $x' \in g_i$. For a typical set $A = \left\{ \begin{array}{ccc} a_1^A & a_2^A & \dots \end{array} \right\}$ and its j-th member $a_j^A \in A$, the superscript $*^A$ denotes the membership, and the subscript $*_j$ denotes the identifier. g_i is used below to propose the controller and the protocol, which implies coordinates, but it can be seen in the controller and the protocol that only $\bar{g}_i$ below is used.

$$\bar{g}_i = \left\{ \begin{array}{ccc} x_1^{g_i} - x_i & x_2^{g_i} - x_i & \dots \end{array} \right\}.$$

where $x' - x_i \in \bar{g}_i$ for any $x' \in g_i$. As $\bar{g}_i$ can be provided by the displacement sensors, data communication is avoided.

Through its coordination protocol, robot x_i needs to make mapping decisions from robots in g_i to points in F, which is difficult since no communication channel is available and the robots could not exchange information. There is, however, a solution to this difficulty. As the limited sensing ranges lead to the intersections of in-range sensed areas among the robots, measurements from these shared areas can be used to make local agreements. Thus, through g_i, we need to find different groups of robots inside the intersected in-range areas.

For robot x_i, the groups s_ι^i of communal neighbours are gathered to be s_i below.

$$\begin{aligned} s_i &= \left\{ \, s_1^i \quad s_2^i \quad \dots \quad s_{\rho_i}^i \, \right\}, \\ s_\iota^i &= \left\{ \, x_1^{s_\iota^i} \quad x_2^{s_\iota^i} \quad x_3^{s_\iota^i} \, \right\}, \end{aligned}$$

where for any $x' \in g_i$ and any $s_\iota^i \in s_i$, $x' \in s_\iota^i$ if

i) $|x' - x''| \leq R$ for any $x'' \in s_\iota^i$ and

ii) for any $\bar{x}' \in g_i \ominus s_\iota^i$, there exists $x''' \in s_\iota^i$

such that $|\bar{x}' - x'''| \geq |x' - x''|$ for any $x', x'' \in s_\iota^i$.

Based on s_ι^i, we also define the communal group reference m_j^i and the set w_j^i, which is a collection of s_ι^i that can access the same m_j^i.

$$\begin{aligned} m_j^i &= \left\{ \, x_1^{m_j^i} \quad x_2^{m_j^i} \quad \dots \, \right\}, \; m_i = \left\{ \, m_1^i \quad m_2^i \quad \dots \, \right\}, \\ w_j^i &= \left\{ \, s_1^{w_j^i} \quad s_2^{w_j^i} \quad \dots \, \right\}, \; w_i = \left\{ \, w_1^i \quad w_2^i \quad \dots \, \right\}, \end{aligned} \tag{3.4}$$

where

iii) for any $x' \in G_i$, $x' \in m_j^i$ if $|x' - x''| \leq R$ for any $x'' \in m_j^i$;

iv) for any $s_\iota^i \in s^i$ and any $w_j^i \in w_i$, $s_\iota^i \in w_j^i$ if $s_\iota^i \subseteq m_j^i$.

By iv), it can be seen that w_j^i includes groups s_ι^i of mutually neighbouring robots that can access the same set of displacements from m_j^i. Therefore, m_j^i can be used as a communal group reference to make mapping decisions among the groups s_ι^i in w_j^i. Thus, by i) and ii), s_ι^i can be seen as a unit to make mapping decisions by Protocol 3.5.1. After the mapping decisions become available, the robots generate the control forces in (3.8) of Section 3.3.

3.2.2 Design of Displacement-Based Reference Formation

In order that the objective formation in (3.2) can be scalable, the size of F needs to be infinite, which means heavy computation burden. However, as only displacements from F are needed for the robots to reach a desired formation, we define a class of finite displacement formations $\Delta\Upsilon$ as the reference for the robot team in (3.1). Based on R and F, the limited displacement formation Υ is defined below.

$$\begin{aligned} \Upsilon &= \left\{ \, \upsilon_1 \quad \upsilon_2 \quad \dots \, \right\}, \\ \Pi &= \left\{ \, \pi_1 \quad \pi_2 \quad \dots \, \right\}, \end{aligned} \tag{3.5}$$

where for any d' and $d'' \in F$, there exist $\pi_i \in \Pi$ and $\upsilon_i \in \Upsilon$ such that $\pi_i = \{\ d' \quad d''\ \}$ and $\upsilon_i = d' - d''$, if $|d' - d''| \leq R$.

Within the sensing range, each robot has neighbours. These groups of neighbours have intersections, within which the robots shared their measurements among each other. As the robots have no fixed positions in F and they need to make suitable mapping decisions, we should define groups of neighbouring points in F to be the robots' mapping counterparts. Under the sensing range R and inside F, we divide the displacements inside Υ into different groups based on the neighbouring points. Thus, based on Υ and R, we define the set $\Delta\Upsilon$ of the displacement formations below.

$$\begin{aligned} \Delta\Upsilon &= \{\ \Delta_1\Upsilon \quad \Delta_2\Upsilon \quad \ldots\ \},\ \Delta_j\Upsilon = \{\ \Delta_1^j\Upsilon \quad \Delta_2^j\Upsilon \quad \Delta_3^j\Upsilon\ \} \subseteq \Upsilon, \\ \Delta\Pi &= \{\ \Delta_1\Pi \quad \Delta_2\Pi \quad \ldots\ \},\ \Delta_j\Pi = (\Delta_1^j\Pi \oplus \Delta_2^j\Pi \oplus \Delta_3^j\Pi) \subseteq F, \end{aligned} \tag{3.6}$$

where

i) for any d' and $d'' \in F$, there exist $\Delta_l^j\Upsilon \in \Delta_j\Upsilon$ and $\Delta_l^j\Pi \in \Delta_j\Pi$ such that $\Delta_l^j\Pi = \{\ d' \quad d''\ \}$ and $\Delta_l^j\Upsilon = d' - d''$ if $|d' - d''| \leq R$;

ii) for any $\Delta_1^j\Upsilon, \Delta_2^j\Upsilon, \Delta_3^j\Upsilon \in \Delta\Upsilon$ and $\Delta_1^j\Pi, \Delta_2^j\Pi, \Delta_3^j\Pi \in \Delta\Pi$, there exist $\Delta_j\Upsilon$ and $\Delta_j\Pi$ such that $\Delta_j\Upsilon = \{\ \Delta_1^j\Upsilon \quad \Delta_2^j\Upsilon \quad \Delta_3^j\Upsilon\ \}$

and

$\Delta_j\Pi = (\Delta_1^j\Pi \oplus \Delta_2^j\Pi \oplus \Delta_3^j\Pi)$ if we have

$$\begin{aligned} &(a) \quad \Delta_1^j\Upsilon + \Delta_2^j\Upsilon + \Delta_3^j\Upsilon = 0, \\ &(b) \quad \Delta_1^j\Pi \odot \Delta_2^j\Pi \odot \Delta_3^j\Pi = \emptyset, \\ &(c) \quad \text{there are only three points in any } \Delta_j\Pi \in \Delta\Pi, \text{ and} \\ &(d) \quad \text{for any } \bar{d}' \in F \ominus \Delta_j\Pi, \text{ there is } d' \in \Delta_j\Pi \\ &\qquad \text{such that } |\bar{d}' - d'| \geq |d'' - d'''| \text{ for any } d'', d''' \in \Delta_j\Pi. \end{aligned}$$

It can be seen that for any $\Delta_l^j\Upsilon \in \Delta_j\Upsilon \in \Delta\Upsilon$ and $\Delta_l^j\Pi \in \Delta_j\Pi \in \Delta\Pi$, there exist $\upsilon_i \in \Upsilon$ and $\pi_i \in \Pi$ such that $\Delta_l^j\Upsilon = \upsilon_i$ and $\Delta_l^j\Pi = \pi_i$. This means that $\Delta\Upsilon$ and $\Delta\Pi$ can be derived from Υ and Π.

The displacement-based framework is used to introduce Υ, Π, $\Delta\Upsilon$, and $\Delta\Pi$. It can be seen that if the sensing range is large enough, then we can derive Υ, Π, $\Delta\Upsilon$, and $\Delta\Pi$ from F, and reconstruct F using either Υ and Π, or $\Delta\Upsilon$ and $\Delta\Pi$. For a randomly generated formation F with an infinite size, the sizes of Υ and $\Delta\Upsilon$ are also infinite. However, if F is generated according to specific rules, Υ and $\Delta\Upsilon$ could have finite sizes. Finiteness with reasonable sizes is feasible for real-time computations and limited data storage capability. If the sizes of Υ $\Delta\Upsilon$ are finite under finite R, then F is a regular scalable formation. We only consider regular scalable formation below. The displacement-based formation Π and $\Delta\Pi$ and the reference formation F is used in Protocol 3.5.1, which is introduced in Section 3.5.

To capture the dynamical behaviour of the robot formation, we model the robots in a team of n robots by the first-order dynamic equation as

$$\begin{cases} x_i(k+1) = x_i(k) + u_i(k), \\ \quad x(k) = \left\{ \begin{array}{cccc} x_1(k) & x_2(k) & \ldots & x_n(k) \end{array} \right\}, \\ \quad u(k) = \left\{ \begin{array}{cccc} u_1(k) & u_2(k) & \ldots & u_n(k) \end{array} \right\}, \end{cases} \tag{3.7}$$

where $x_i(k)$, $i = 1, \ldots, n$ is the position of robot x_i at instant k, $u_i(k)$ is its control force, and $x(k) = [\ x_1^T(k) \ \ \ldots \ \ x_n^T(k)\ ]^T$ is the robot formation at k-th instant. The overall control goal is to drive all the robots to the displacements from the objective formation F.

3.3 Controller Design and Analysis

It is assumed that each robot can only measure the displacements to others within the radius R, that data communication channels are not available, and that the robots are indistinguishable for the sensors. This can lead to the classical controllers not being practical. We design a controller that does not rely on data communication and drives all robots to the displacements from the objective formation. We show that due to the localisation, the robots in the formation team are subject to potential mapping collisions. To overcome the problem, in Section 3.5, a coordination protocol is proposed to avoid the collisions while maintaining connectivity.

As the robots do not have fixed positions in the objective formation, they have to make real-time mapping decisions and such decisions can only be based on a set of local information. Therefore, under different sets of local information, there may be multiple decisions on one robot's mapping position in the objective formation. As a result, when the robots are still in the process of finding the most suitable mapping position, mappings for the moment are not very trustworthy. This means that a bigger error doesn't deserve a bigger influence in the control force.

We embody such consideration by the normalised sum. Using the displacement among the robots and the displacements among the mappings, the dynamics of robot x_i under control force $u_i(k)$ is

$$\left\{ \begin{aligned} x_i(k+1) &= x_i(k) + u_i(k), \\ u_i(k) &= \sum_{s_\iota^i \in s_i} f_j^i(k), \\ f_j^i(k) &= -\tau \sum_{x_i, x' \in s_\iota^i}^{x' \neq x_i} \tfrac{x_i(k) - x'(k) - d^{s_\iota^i}(x_i(k)) + d^{s_\iota^i}(x'(k))}{|x_i(k) - x'(k) - d^{s_\iota^i}(x_i(k)) + d^{s_\iota^i}(x'(k))|}, \end{aligned} \right. \tag{3.8}$$

where τ is the sampling period. The mapping from $x' \in s_\iota^i$ to $d^{s_\iota^i}(x') \in F$ is a decision made by the robots in s_ι^i, and such a decision is accessible to all the robots in s_ι^i. The unanimous decision is possible because s_ι^i is a group of communal neighbours and the robots in s_ι^i share the same communal group reference $m_{j'}^i$, which is a set of local information for robots inside s_ι^i. In Section 3.5, based on $m_{j'}^i$, $\Delta\Pi$, and Π, the coordination protocol is designed to provide the mappings $d^{s_\iota^i}(x_i(k))$, $d^{s_\iota^i}(x'(k))$.

If all the mapping decisions are gathered, we can have $d(s^i)$ and $d(g_i)$ below.

$$\begin{aligned} d(s_i) &= \left\{\ d(s_1^i) \quad d(s_2^i) \quad \ldots\ \right\}, \\ d(g_i) &= \left\{\ d^{g_i}(x_1^{g_i}) \quad d^{g_i}(x_2^{g_i}) \quad \ldots\ \right\}, \end{aligned} \tag{3.9}$$

$$\begin{aligned} d(s_\iota^i) &= \left\{\ d^{s_\iota^i}(x_1^{s_\iota^i}) \quad d^{s_\iota^i}(x_2^{s_\iota^i}) \quad d^{s_\iota^i}(x_3^{s_\iota^i})\ \right\}; \\ d^{g_i}(x_q^{g_i}) &= \left\{\ d^{s_1^i}(x_q^{g_i}) \quad d^{s_2^i}(x_q^{g_i}) \quad \ldots\ \right\}, \end{aligned} \tag{3.10}$$

where $d^{s_\iota^i}(x_l^{s_\iota^i}) \in F$ and $d(s_\iota^i) \in \Delta\Pi$ for any $x_l^{s_\iota^i} \in s_\iota^i$. The mappings from $x_l^{s_\iota^i} \in s_\iota^i \subseteq x$ to $d^{s_\iota^i}(x_l^{s_\iota^i}) \in F$ and from $s_\iota^i \subseteq g_i$ to $d(s_\iota^i) \in \Delta\Pi$ are provided by the protocol in Section 3.5.

As all the robots in $s_\iota^i \subseteq m_j^i$ have access to m_j^i in (3.4), the mapping decision $d(s_\iota^i)$ can be made based on m_j^i. Moreover, as all the $s_1^{w_j^i}, s_2^{w_j^i}, \ldots$ and s_ι^i in w_j^i can access the same m_j^i, the involved robots could reach a unanimous mapping decision. However, for a robot $x_q^{g_i}$ in g_i, it may belong to different groups $s_1^i, s_2^i, \ldots$. It is possible that some of $s_1^i, s_2^i, \ldots$ belong to different $w_1^i, w_2^i, \ldots$ and have access to different $m_1^i, m_2^i, \ldots$. Thus, they could have multiple mappings $d^{s_1^i}(x_q^{g_i}), d^{s_2^i}(x_q^{g_i}), \ldots$. If some of the mappings are different, then we say there there are mapping collisions. Next, we analyse the formation performance in the presence of mapping collisions.

We analyse the performance of the controller in (3.8). We call $\delta_{s_\iota^i,l}(k)$ a control edge from robot $x_i \in s_\iota^i$ to robot $x_l^{s_\iota^i} \in s_\iota^i$, where

$$\delta_{s_\iota^i,l}(k) = x_i(k) - x_l^{s_\iota^i}(k) - d^{s_\iota^i}(x_i(k)) + d^{s_\iota^i}(x_l^{s_\iota^i}(k)), \ \text{ for } \ l = 1, 2, 3.$$

Gather a group of control edges and denote

$$\delta_{s_i}(k) = [\ \delta^T_{s^i_1}(k) \quad \delta^T_{s^i_2}(k) \quad \dots\ ]^T,$$

where

$$\delta_{s^i_\iota}(k) = \left[\ \delta^T_{s^i_\iota,1}(k) \quad \delta^T_{s^i_\iota,2}(k) \quad \delta^T_{s^i_\iota,3}(k)\ \right]^T.$$

It can be seen that $\delta_{s^i_\iota}(k)$ is produced from s^i_ι, and δ_{s_i} is produced from s_i. Thus, $\delta_{s_i}(k)$ takes full advantage of g_i. As the normalised sum is used to design the control forces, there exist $\mathcal{H}_{s^i_\iota,1}(k)$ and $\mathcal{H}_{s^i_\iota,1}(k)$ such that the normalised counterpart $\bar{\delta}_{s_i}(k)$ of δ_{s_i} is defined below.

$$\bar{\delta}_{s_i}(k) = [\ \bar{\delta}^T_{s^i_1}(k) \quad \bar{\delta}^T_{s^i_2}(k) \quad \dots\ ]^T,$$

where

$$\begin{aligned} \bar{\delta}_{s^i_\iota}(k) &= \left[\ \bar{\delta}_{s^i_\iota,1}(k) \quad \bar{\delta}_{s^i_\iota,2}(k)\ \right]^T; \\ \bar{\delta}_{s^i_\iota,l}(k) &= \frac{\mathcal{H}_{s^i_\iota,l}(k)\delta_{s^i_\iota}(k)}{|\mathcal{H}_{s^i_\iota,l}(k)\delta_{s^i_\iota}(k)|}, \text{ for } l = 1, 2; \end{aligned}$$

For $\mathcal{H}_{s^i_\iota,1}(k)$ and $\mathcal{H}_{s^i_\iota,1}(k)$, we have

i) $\mathcal{H}_{s^i_\iota,l}(k) \left[\ \delta^T_{s^i_\iota,1}(k) \quad \delta^T_{s^i_\iota,2}(k) \quad \delta^T_{s^i_\iota,3}(k)\ \right]^T \in \left\{\ \delta_{s^i_\iota,1}(k) \quad \delta_{s^i_\iota,2}(k) \quad \delta_{s^i_\iota,3}(k)\ \right\}$;

ii) $\mathcal{H}_{s^i_\iota,1}(k) \neq \mathcal{H}_{s^i_\iota,2}(k)$, $\mathcal{H}_{s^i_\iota,1}(k) \neq 0$, and $\mathcal{H}_{s^i_\iota,2}(k) \neq 0$,

where $l = 1, 2$. If the $\delta_{s_i}(k)$ and $\bar{\delta}_{s_i}(k)$ from all the robots x_i, $i = 1, 2, \dots, n$ are gathered, we have

$$\begin{aligned} \delta(k) &= [\ \delta^T_{s_1}(k) \quad \delta^T_{s_2}(k) \quad \dots \quad \delta^T_{s_n}(k)\ ]^T, \\ \bar{\delta}(k) &= [\ \bar{\delta}^T_{s_1}(k) \quad \bar{\delta}^T_{s_2}(k) \quad \dots \quad \bar{\delta}^T_{s_n}(k)\ ]^T. \end{aligned}$$

As the mapping decision $d(s^i_\iota)$ is made based on $m^i_{j'}$, a group of robots that robot groups in w^i_j can access, the decision would be unanimous in s^i_ι and among the robot groups in w^i_j. Thus, it can be seen that given that x' and $x'' \in s^i_\iota$, and $\delta_{s^i_\iota,l}(k)$ and $\delta_{s^i_\iota,l'}(k)$ are the control edges from robot x' to x'' and from x'' to x', we have

$$\delta_{s^i_\iota,l}(k) = -\delta_{s^i_\iota,l'}(k).$$

Thus, $\delta(k)$ is not linearly independent. there exists $P(k)$ such that we can retain one of such $\delta_{s^i_\iota,l}(k)$ and $\delta_{s^i_\iota,l'}(k)$ and generate new $\hat{\delta}(k)$ and $\bar{\hat{\delta}}(k)$ from $\delta(k)$ and $\bar{\delta}(k)$ by

$$\delta(k) = P(k)\hat{\delta}(k), \quad \bar{\delta}(k) = P(k)\bar{\hat{\delta}}(k).$$

Then, we have

$$\begin{aligned}\bar{\delta}^T(k)\bar{\delta}(k) &= n = 2 \times \frac{n}{2}\\ &= 2\bar{\hat{\delta}}^T(k)\bar{\hat{\delta}}(k) = \bar{\hat{\delta}}^T(k)P^T(k)P(k)\bar{\hat{\delta}}(k).\end{aligned}$$

Since $\bar{\hat{\delta}}(k)$ is linearly independent, we have

$$P^T(k)P(k) = 2I,$$

which means that $P(k)$ is full column rank.

Although the form expressed in (3.8) is intuitive, it is not well suited for formation analysis. A better perspective is to rewrite by introducing

$$E(k) = \left[\begin{array}{ccc} e_{m_1(k)} & \mathbf{0}_{\sum_{s=2}^{n} m_s(k)} & \\ \ddots & \ddots & \\ \mathbf{0}_{\sum_{s=1}^{l-1} m_s(k)} & e_{m_l(k)} & \mathbf{0}_{\sum_{s=l+1}^{n} m_s(k)} \\ \ddots & & \ddots \\ & \mathbf{0}_{\sum_{s=1}^{n-1} m_s(k)} & e_{m_n(k)} \end{array}\right],$$

where $X(k) = [\ x_1 \quad x_2 \quad ...\ ]$, $e_l = [\ 1 \quad ... \quad 1\ ]$ and $\mathbf{0}_l = [\ 0 \quad ... \quad 0\ ]$ are lth. By (3.8), we have

$$X(k+1) = X(k) - \tau E(k)P(k)\bar{\hat{\delta}}(k), \tag{3.11}$$

which is more suitable for analysing formation control. Note that $P(k)$ and $E(k)$ are dependent on the formation topology. If the topology of the robot formation stays the same and the mapping decisions are not changed from instant k to $k+1$, then

$$\begin{aligned} P(k) &= P(k+1),\\ E(k) &= E(k+1),\\ d^{s_\iota^i}(x'(k+1)) &= d^{s_\iota^i}(x'(k)), \text{ for any } x' \in s_\iota^i \in s_i.\end{aligned}$$

Consequently, by (3.11), we have

$$\begin{aligned}\hat{\delta}(k+1) - \hat{\delta}(k) &= P^T(k)E^T(k)\left(X(k+1) - X(k)\right)\\ &= -\tau P^T(k)E^T(k)E(k)P(k)\bar{\hat{\delta}}(k),\\ \hat{\delta}(k+1) &= \hat{\delta}(k) - \tau P^T(k)E^T(k)E(k)P(k)\bar{\hat{\delta}}(k). \end{aligned}\tag{3.12}$$

As the control forces are normalised, the steady state is related to the sampling time τ. To derive the restriction on τ, we decouple the system matrix. Suppose that there exist $U(k)$, $\Lambda_{\hat{\delta}}(k)$, and $\Lambda^*_{\hat{\delta}}(k)$ such that

$$\Lambda_{\hat{\delta}}(k) = U(k)P^T(k)E^T(k)E(k)P(k)U^T(k) \leq \Lambda^*_{\hat{\delta}}(k).$$

where

$$\begin{aligned} U(k)U^T(k) &= I, \\ \Lambda_{\hat{\delta}}(k) &= \mathrm{diag}(\lambda_1, \ldots, \lambda_{\sum_{s=1}^n m_s(k)}), \\ \Lambda^*_{\hat{\delta}}(k) &= \mathrm{diag}(\lambda^*_1, \ldots, \lambda^*_{\sum_{s=1}^n m_s(k)}). \end{aligned}$$

There exists $R(\hat{\delta}(k))$ such that $\bar{\hat{\delta}}(k) = R(\hat{\delta}(k))\hat{\delta}(k)$. As $\bar{\hat{\delta}}(k)$ is normalised and $\hat{\delta}$ is a vector of control edges, $R(\hat{\delta}(k))$ is a diagonal matrix, whose diagonal elements are the reciprocals of the lengths of the control edges (we call the lengths of control edges *misalignment errors*). Then,

$$\begin{aligned} & U(k)\left(\hat{\delta}(k+1) - \hat{\delta}(k)\right) \\ &= -\tau U(k)P^T(k)E^T(k)E(k)P(k)\bar{\hat{\delta}}(k) \\ &= -\tau U(k)P^T(k)E^T(k)E(k)P(k)U^{-1}(k)U(k) \times R(\hat{\delta}(k))\hat{\delta}(k) \\ &= -\tau U(k)P^T(k)E^T(k)E(k)P(k)U^{-1}(k) \times R(\hat{\delta}(k))U(k)\hat{\delta}(k) \\ &= -\tau \Lambda_{\hat{\delta}}(k)R(\hat{\delta}(k))U(k)\hat{\delta}(k). \end{aligned}$$

Assume that

$$\hat{\delta}'(k) = U(k)\hat{\delta}(k), \;\; U(k) = U(k+1).$$

We have

$$\begin{aligned} \hat{\delta}'(k+1) &= \hat{\delta}'(k) - \tau\Lambda_{\hat{\delta}}(k)R(\hat{\delta}(k))\hat{\delta}'(k) \\ &= (I - \tau\Lambda_{\hat{\delta}}(k)R(\hat{\delta}(k)))\hat{\delta}'(k) \\ &= S(\hat{\delta}(k))\hat{\delta}'(k). \end{aligned} \tag{3.13}$$

Large sampling period leads to large misalignment error, which could enlarge the inter-robot distances and possibly break links among the robots. Thus, we need to use the decoupled form (3.13) to derive the misalignment error $\delta(k)$ under a fixed sampling period without mapping collisions. Moreover, we can derive the sampling period to achieve a tolerable misalignment error. For ensuring the convergence of the robot formation, the condition below is proposed.

$$0 \leq \tau\Lambda_{\hat{\delta}}(k)R(\hat{\delta}(k)) \leq I.$$

Thus, if the maximum eigenvalues of $\Lambda_{\hat{\delta}}(k)$ and $R(\hat{\delta}(k))$ are λ^*_{max} and $\frac{1}{|\delta_{max}(k)|}$, we then choose $\tau \leq \frac{|\delta_{max}(k)|}{\lambda^*_{max}}$. Also, given a fixed τ, and with λ^*_{max} and $\frac{1}{|\delta_{min}(k)|}$ as the minimum positive eigenvalues of $\Lambda_{\hat{\delta}}(k)$ and $R(\hat{\delta}(k))$, the largest misalignment error is $|\delta_{min}(k)| \leq \tau\lambda^*_{min}$.

If the sum of the control forces is zero, then the robots have reached the steady states. We analyse the stability of the robot formation. Assume that $W(k)$ is symmetric non-negative definite and $W(k+1) = W(k)$. We have

$$\begin{aligned}
&\hat{\delta}^T(k+1)W(k+1)\hat{\delta}(k+1) - \hat{\delta}^T(k)W(k)\hat{\delta}(k)\\
= &-\hat{\delta}^T(k)\left(W(k) - S^T(\hat{\delta}(k))W(k)S(\hat{\delta}(k))\right)\hat{\delta}(k) \leq 0,\\
&\hat{\delta}^T(k+1)W(k+1)\hat{\delta}(k+1) \leq \hat{\delta}^T(k)W(k)\hat{\delta}(k).
\end{aligned}$$

Since $\hat{\delta}(k) \to \hat{\delta}(k+1)$ as $k \to \infty$, using (3.12), we have

$$P^T(\infty)E^T(\infty)E(\infty)P(\infty)\bar{\hat{\delta}}(\infty) = 0. \tag{3.14}$$

Members of $\bar{\hat{\delta}}(\infty)$ are not necessarily zero by (3.14). Through the analysis below, we show some important property of the steady states.

3.4 Mapping Collisions in the Robot Formation

We have designed a novel controller (3.8). Below, we show that it leads to potential nonzero misalignment errors in the presence of mapping collisions. As the mapping collisions are possibly present, there exist $\tilde{\delta}(k)$, $d_\epsilon \neq 0$, $R_{\tilde{\delta}}(k)$, and $R_{d_\epsilon}(k)$ such that

$$\begin{aligned}
\hat{\delta}(k) &= R_{\tilde{\delta}}(k)\tilde{\delta}(k) + R_{d_\epsilon}(k)d_\epsilon(k),\\
\tilde{\delta}(k) &= \left[\begin{array}{ccc} \tilde{\delta}^T_1(k) & ... & \tilde{\delta}^T_{n-1}(k) \end{array}\right]^T, d_\epsilon(k) = \left[\begin{array}{ccc} d^T_{\epsilon_1}(k) & ... & d^T_{\epsilon_\kappa}(k) \end{array}\right]^T.
\end{aligned} \tag{3.15}$$

Thus, there should exist $R^e_{\tilde{\delta}}(k)$ and $R^e_{d_\epsilon}(k)$ such that

$$R(\hat{\delta})\hat{\delta}(k) = R_{\tilde{\delta}}(k)R^e_{\tilde{\delta}}(k)R(\tilde{\delta})\tilde{\delta}(k) + R_{d_\epsilon}(k)R^e_{d_\epsilon}(k)R(d_\epsilon)(k)d_\epsilon(k). \tag{3.16}$$

To derive the non-zero steady state error, by (3.16), we write the energy function $\hat{\delta}^T(k)\hat{\delta}(k)$ as

$$\begin{aligned}\hat{\delta}^T(k)\hat{\delta}(k) &= \Big(R_{\tilde{\delta}}(k)\tilde{\delta}(k) + R_{d_\epsilon}(k)d_\epsilon(k)\Big)^2 \\ &= \tilde{\delta}^T(k)R_{\tilde{\delta}}^T(k)R_{\tilde{\delta}}(k)\tilde{\delta}(k) + d_\epsilon^T(k)R_{d_\epsilon}^T(k)R_{d_\epsilon}(k)d_\epsilon(k) \\ &\quad + \ \tilde{\delta}^T(k)R_{\tilde{\delta}}^T(k)R_{d_\epsilon}(k)d_\epsilon(k) + d_\epsilon^T(k)R_{d_\epsilon}^T(k)R_{\tilde{\delta}}(k)\tilde{\delta}(k).\end{aligned}$$

If $\tilde{\delta}(\infty) = 0$, we have

$$\hat{\delta}^T(k)\hat{\delta}(k) \to d_\epsilon^T(\infty)R_{d_\epsilon}^T(\infty)R_{d_\epsilon}(\infty)d_\epsilon(\infty).$$

Since $d_\epsilon \neq 0$, $\hat{\delta}^T(k)\hat{\delta}(k)$ can not converge to zero. This means that the control law (3.8) could lead to a partially desired formation. If k is sufficiently large and τ is sufficiently small, then

$$\hat{\delta}(k+1) \approx \hat{\delta}(k).$$

Substitute (3.15) and (3.16) into (3.12), we have

$$\begin{aligned}&\hat{\delta}^T(k+1)\hat{\delta}(k+1) - \hat{\delta}^T(k)\hat{\delta}(k) \\ =& \Big(\hat{\delta}(k+1) + \hat{\delta}(k)\Big)^T \Big(\hat{\delta}(k+1) - \hat{\delta}(k)\Big) \\ \approx& -2\tau\hat{\delta}^T(k)P^T(k)E^T(k)E(k)P(k)\bar{\hat{\delta}}(k) \\ =& -2\tau\left(R_{\tilde{\delta}}(k)\tilde{\delta}(k) + R_{d_\epsilon}(k)d_\epsilon\right)^T(k)P^T(k)E^T(k) \times E(k)P(k) \\ & \times\left(R_{\tilde{\delta}}(k)R_{\tilde{\delta}}^e(k)R(\tilde{\delta}(k))\tilde{\delta}(k) + R_{d_\epsilon}(k)R_{d_\epsilon}^e(k)R(d_\epsilon(k))d_\epsilon(k)\right) \\ =& -2\tau\left[\begin{array}{c}\tilde{\delta}(k) \\ d_\epsilon(k)\end{array}\right]^T \left[\begin{array}{cc}R_{\tilde{\delta}}(k) & R_{d_\epsilon}(k)\end{array}\right]^T P^T(k) \times E^T(k)E(k)P(k) \\ & \times\left[\begin{array}{cc}R_{\tilde{\delta}}(k) & R_{d_\epsilon}(k)\end{array}\right]\left[\begin{array}{cc}R_{\tilde{\delta}}^e(k) & 0 \\ 0 & R_{d_\epsilon}^e(k)\end{array}\right]\left[\begin{array}{c}R(\tilde{\delta}(k))\tilde{\delta}(k) \\ R(d_\epsilon(k))d_\epsilon(k)\end{array}\right] \\ =& -2\tau\left(\left[\begin{array}{cc}R_{\tilde{\delta}}^e(k) & 0 \\ 0 & \left[\begin{array}{cc}(R_{d_\epsilon}^e)_1(k) & 0 \\ 0 & (R_{d_\epsilon}^e)_2(k)\end{array}\right]\end{array}\right] \times \left[\begin{array}{c}R(\tilde{\delta}(k))\tilde{\delta}(k) \\ R(d_\epsilon(k))d_\epsilon(k)\end{array}\right]\right)^T \\ & \times\left[\begin{array}{ccc}I & 0 & 0 \\ 0 & (R_{d_\epsilon}^e)_1^{-T}(k) & 0 \\ 0 & 0 & (R_{d_\epsilon}^e)_2^{-T}(k)\end{array}\right]\end{aligned}$$

$$
\begin{aligned}
&\times \begin{bmatrix} (R^e_{\tilde{\delta}})^{-T}(k)R(\tilde{\delta}(k)) & 0 & 0 \\ 0 & (R(d_\epsilon))_1(k) & 0 \\ 0 & 0 & (R(d_\epsilon))_2(k) \end{bmatrix} \\
&\times \begin{bmatrix} R_{\tilde{\delta}}(k) & R_{d_\epsilon}(k) \end{bmatrix}^T P^T(k)E^T(k)E(k)P(k) \\
&\times \begin{bmatrix} R_{\tilde{\delta}}(k) & R_{d_\epsilon}(k) \end{bmatrix} \left(\begin{bmatrix} R^e_{\tilde{\delta}}(k) & 0 \\ 0 & \begin{bmatrix} (R^e_{d_\epsilon})_1(k) & 0 \\ 0 & (R^e_{d_\epsilon})_2(k) \end{bmatrix} \end{bmatrix} \right. \\
&\left. \times \begin{bmatrix} R(\tilde{\delta}(k))\tilde{\delta}(k) \\ R(d_\epsilon(k))d_\epsilon(k) \end{bmatrix} \right) \\
= -2\tau &\left(\begin{bmatrix} R^e_{\tilde{\delta}}(k) & 0 \\ 0 & \begin{bmatrix} (R^e_{d_\epsilon})_1(k) & 0 \\ 0 & I \end{bmatrix} \end{bmatrix} \begin{bmatrix} R(\tilde{\delta}(k))\tilde{\delta}(k) \\ R(d_\epsilon(k))d_\epsilon(k) \end{bmatrix} \right)^T \\
&\times \begin{bmatrix} I & 0 & 0 \\ 0 & (R^e_{d_\epsilon})_1^{-T}(k) & 0 \\ 0 & 0 & I \end{bmatrix} \\
&\times \begin{bmatrix} (R^e_{\tilde{\delta}})^{-T}(k)R(\tilde{\delta}(k)) & 0 & 0 \\ 0 & (R(d_\epsilon))_1(k) & 0 \\ 0 & 0 & (R(d_\epsilon))_2(k) \end{bmatrix} \begin{bmatrix} R_{\tilde{\delta}}(k) & R_{d_\epsilon}(k) \end{bmatrix}^T \\
&\times P^T(k)E^T(k)E(k)P(k) \begin{bmatrix} R_{\tilde{\delta}}(k) & R_{d_\epsilon}(k) \end{bmatrix} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & (R^e_{d_\epsilon})_2(k) \end{bmatrix} \\
&\times \left(\begin{bmatrix} R^e_{\tilde{\delta}}(k) & 0 \\ 0 & \begin{bmatrix} (R^e_{d_\epsilon})_1(k) & 0 \\ 0 & I \end{bmatrix} \end{bmatrix} \begin{bmatrix} R(\tilde{\delta}(k))\tilde{\delta}(k) \\ R(d_\epsilon(k))d_\epsilon(k) \end{bmatrix} \right) \leq 0,
\end{aligned}
$$

where

$$
\begin{aligned}
R^e_{d_\epsilon}(k) &= \begin{bmatrix} (R^e_{d_\epsilon})_1(k) & 0 \\ 0 & (R^e_{d_\epsilon})_2(k) \end{bmatrix}, \\
R(d_\epsilon(k)) &= \begin{bmatrix} (R(d_\epsilon))_1(k) & 0 \\ 0 & (R(d_\epsilon))_2(k) \end{bmatrix}.
\end{aligned}
$$

Since every element of

$$
\begin{bmatrix} R^e_{\tilde{\delta}}(\infty) & 0 \\ 0 & \begin{bmatrix} (R^e_{d_\epsilon})_1(\infty) & 0 \\ 0 & I \end{bmatrix} \end{bmatrix} \begin{bmatrix} R(\tilde{\delta}(k))\tilde{\delta}(\infty) \\ R(d_\epsilon(k))d_\epsilon(\infty) \end{bmatrix}
$$

is nonzero and communally exclusive, similar with (3.14), we have

$$\begin{aligned}&\begin{bmatrix} I & 0 & 0 \\ 0 & (R^e_{d_\epsilon})_1^{-T}(\infty) & 0 \\ 0 & 0 & I \end{bmatrix}\\ &\times \begin{bmatrix} (R^e_{\tilde{\delta}})^{-T}(\infty)R(\tilde{\delta}(\infty)) & 0 & 0 \\ 0 & (R(d_\epsilon))_1(\infty) & 0 \\ 0 & 0 & (R(d_\epsilon))_2(\infty) \end{bmatrix}\\ &\times \begin{bmatrix} R_{\tilde{\delta}}(\infty) & R_{d_\epsilon}(\infty) \end{bmatrix}^T P^T(\infty)E^T(\infty)E(\infty)P(\infty)\\ &\times \begin{bmatrix} R_{\tilde{\delta}}(\infty) & R_{d_\epsilon}(\infty) \end{bmatrix} \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & (R^e_{d_\epsilon})_2(\infty) \end{bmatrix} = 0.\end{aligned}$$

Our deliberate selection makes $R(\tilde{\delta}(\infty))$ equal to zero and by (3.16), $R^e_{d_\epsilon}(\infty)$ is partially zero and the reciprocal of its other part is zero. Since $(R^e_{d_\epsilon})_1^{-T}(\infty)$ and $(R^e_{d_\epsilon})_2(\infty)$ are zero, then

$$\begin{aligned}&\begin{bmatrix} I & 0 & 0 \\ 0 & (R^e_{d_\epsilon})_1^{-T}(\infty) & 0 \\ 0 & 0 & I \end{bmatrix}\\ &\times \begin{bmatrix} (R^e_{\tilde{\delta}})^{-T}(\infty)R(\tilde{\delta}(\infty)) & 0 & 0 \\ 0 & (R(d_\epsilon))_1(\infty) & 0 \\ 0 & 0 & (R(d_\epsilon))_2(\infty) \end{bmatrix}\\ &\times \begin{bmatrix} R_{\tilde{\delta}}(\infty) & R_{d_\epsilon}(\infty) \end{bmatrix}^T P^T E^T E P \begin{bmatrix} R_{\tilde{\delta}}(\infty) & R_{d_\epsilon}(\infty) \end{bmatrix} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & * \end{bmatrix},\\ &* \neq 0,\\ &\begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & (R^e_{d_\epsilon})_2(\infty) \end{bmatrix} = \begin{bmatrix} I & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & 0 \end{bmatrix}.\end{aligned}$$

That is,

$$\begin{bmatrix} (R^e_{\tilde{\delta}})^{-T}(\infty)R(\tilde{\delta}(\infty)) & 0 & 0 \\ 0 & (R^e_{d_\epsilon})_1^{-T}(\infty)(R(d_\epsilon))_1(\infty) & 0 \\ 0 & 0 & (R^e_{d_\epsilon})_2(\infty)(R(d_\epsilon))_2(\infty) \end{bmatrix} = 0.$$

Since $(R(d_\epsilon))_1(\infty)$, $(R(d_\epsilon))_2(\infty) \neq 0$, we have

$$\begin{bmatrix} (R^e_{\tilde{\delta}})^{-T}(\infty)R(\tilde{\delta}(\infty)) & 0 & 0 \\ 0 & (R^e_{d_\epsilon})_1^{-T}(\infty) & 0 \\ 0 & 0 & (R^e_{d_\epsilon})_2(\infty) \end{bmatrix} = 0.$$

In fact, with the ability to choose the control edges in $\tilde{\delta}(\infty)$, we choose a control edge such that $R(\tilde{\delta}(\infty)) = 0$. Thus, at $k = \infty$,

$$\hat{\delta}^T(k)\hat{\delta}(k) = d_\epsilon^T(k)R_{d_\epsilon}^T(k)R_{d_\epsilon}(k)d_\epsilon(k), \ \ \tilde{\delta}^T(k)R_{\tilde{\delta}}^T(k)R_{\tilde{\delta}}(k)\tilde{\delta}(k) = 0.$$

This means that as long as the mapping collisions exist, a residue persists and $\hat{\delta}^T(k)\hat{\delta}(k)$ is always positive. As a consequence, the robot formation cannot fully reach the displacements from the objective formation.

The formation is connected through a path of robots relevant with $(R^e_{d_\epsilon})_2(\infty) = 0$, and the mapping collisions are among robots relevant with $(R^e_{d_\epsilon})_1(\infty) \neq 0$. With this advantage, Protocol 3.5.1 is able to identify the local mapping collisions. Moreover, many studies [18, 26, 29, 32] have focused on second-order models. Changing (3.7) into a second-order robot formation, results in this chapter could also be applied.

3.5 Design of Coordination Protocol

With the access to Π in (3.5) and $\Delta\Pi$ in (3.6) derived from F in (3.2), a single robot x_i needs to make decisions on $d(g_i)$ in (3.9) from g_i in (3.3), a mapping relation which is provided by the coordination protocol. In the preceding analysis, the mapping collisions prevent the robot formation from reaching a desired formation. In this section, by assuming that neighbouring robots should be mapped to neighbouring points in the objective formation, the proposed protocol should be able to avoid the mapping collisions. The concept of exclusiveness below is proposed to define a group of neighbouring robots or points.

Definition 3.5.1. *Given two groups g and g' of points, if for any $\bar{x}' \in g \ominus g'$, there exist x' and $x'' \in g'$ and $x_\perp \neq 0$ such that*

$$x_\perp^T(x' - \bar{x}') \cdot x_\perp^T(x' - x''') \leq 0,$$

where $x_\perp^T(x' - x'') = 0$ *for any* $x''' \in g' \ominus \left\{ x' \quad x'' \right\}$*, then we say* g' *is exclusive in* g*.*

Note that Definition 3.5.1 proposes a criterion to identify a convex set from groups of points. If g' is exclusive in g, then g' is compact such that there exists a convex polygon encircling all the points in g' and any point in $g \ominus g'$ is outside such polygon. Under the control of the scalable formations, exclusiveness is used to avoid infinite mapping loops. Moreover, to avoid mapping collisions, the concept of groups of communal neighbours and compatibility is proposed below.

Definition 3.5.2. *For a group* X' *of robots, if* $|x' - x''| \leq R$ *for any* x' *and* $x'' \in X'$*, then we call* X' *a group of communal neighbours.*

Definition 3.5.3. *For any two groups* A *and* B *of communal neighbours, we say* A *is compatible in* B *if* $d^A(x') - d^A(x'') = d^B(x') - d^B(x'')$ *for any* x' *and* $x'' \in A \odot B$*, where* $d^A(x')$ *and* $d^A(x'')$ *are mappings in* A*, and* $d^B(x')$ *and* $d^B(x'')$ *are mappings in* B*.*

To map a group of robots to a group of points, we need to propose a mapping criterion $L^{X'}$. For a group $X' = \{x'_1, x'_2, \dots\}$ of communal neighbours, we have the formation residue energy

$$L_{X'} = \sum_{x', x'' \in X'} \left[x' - x'' - d^{X'}(x') + d^{X'}(x'') \right]^2, \tag{3.17}$$

where $d^{X'}(x')$ and $d^{X'}(x'')$ are robot x''s and x'''s mappings in X'. Based on exclusiveness, groups of mutual neighbours, compatibility and the formation residue energy $L_{X'}$, the coordination protocol for robot x_i is proposed below.

Protocol 3.5.1. *(A) If there are three robots in* $s^i_\iota \in s^i$ *in (3.4), then robot* x_i *could follow the process below to solve* $d(s^i_\iota)$ *in (3.10) for* s^i_ι*:*

Step 0: The most suitable mapping $d(s^i_{j_1^*})$ *is derived for* $s^i_{j_1^*} \in w^i_l$ *in (3.18).*

$$\begin{aligned} s^i_{j_1^*} &= \arg \min_{s' \in w^i_l} \min_{d(s') \in \Delta\Pi} L_{s'}, \\ d(s^i_{j_1^*}) &= \arg \min_{d(s^i_{j_1^*}) \in \Delta\Pi} L_{s^i_{j_1^*}}. \end{aligned} \tag{3.18}$$

If $s^i_\iota = s^i_{j_1^*}$*, then* $d(s^i_\iota) = d(s^i_{j_1^*})$ *and continue to Step 0 of the next iteration.*
If $s^i_\iota \neq s^i_{j_1^*}$*, then*

(a) $\mathcal{W}_{j_1^*} = \left\{ \bar{x}_1^{\mathcal{W}_{j_1^*}} \quad \bar{x}_2^{\mathcal{W}_{j_1^*}} \quad \ldots \right\}$*, where*

for any $\bar{x} \in m^i_j \ominus s^i_{j_1^*}$*,* $\bar{x} \in \mathcal{W}_{j_1^*}$ *if* $s^i_{j_1^*} \oplus \{\bar{x}\}$ *is exclusive in* m^i_j*;*

(b) $F_{j_1^*} = \left\{ d_1^{F_{j_1^*}} \quad d_2^{F_{j_1^*}} \quad \ldots \right\}$*, where*

for any $d' \in F$*,* $d' \in F_{j_1^*}$ *if* $d(s^i_{j_1^*}) \oplus \{d'\}$ *is exclusive in* F*.*

Since $d(s^i_\iota)$ *is not found, we continue to Step 1 with* $s^i_{j_1^*}$*,* $d(s^i_{j_1^*})$*,* $\mathcal{W}_{j_1^*}$ *and* $F_{j_1^*}$*.*

Step 1: The most suitable mapping $d(\bar{x}_{p_1^*}^{\mathcal{W}_{j_1^*}})$ *is derived for* $\bar{x}_{p_1^*}^{\mathcal{W}_{j_1^*}} \in \mathcal{W}_{j_1^*}$ *in (3.19).*

$$\begin{aligned} \bar{x}_{p_1^*}^{\mathcal{W}_{j_1^*}} &= \arg \min_{\bar{x} \in \mathcal{W}_{j_1^*}} \min_{d(\bar{x}) \in F_{j_1^*}} L_{s^i_{j_1^*} \oplus \{\bar{x}\}}, \\ d(\bar{x}_{p_1^*}^{\mathcal{W}_{j_1^*}}) &= \arg \min_{d(\bar{x}_{p_1^*}^{\mathcal{W}_{j_1^*}}) \in F_{j_1^*}} L_{s^i_{j_1^*} \oplus \left\{ \bar{x}_{p_1^*}^{\mathcal{W}_{j_1^*}} \right\}}. \end{aligned} \tag{3.19}$$

We have $\boldsymbol{s}^i_{j_2^*} = s^i_{j_1^*} \oplus \left\{ \bar{x}_{p_1^*}^{\mathcal{W}_{j_1^*}} \right\}$*.*
If $s^i_\iota \subset \boldsymbol{s}^i_{j_2^*}$*, then* $d(s^i_\iota) \subset d(\boldsymbol{s}^i_{j_2^*})$ *and continue to Step 0 of the next iteration.*
If $s^i_\iota \not\subset \boldsymbol{s}^i_{j_2^*}$*, then*

(a) $\mathcal{W}_{j_2^*} = \left\{ \bar{x}_1^{\mathcal{W}_{j_2^*}} \quad \bar{x}_2^{\mathcal{W}_{j_2^*}} \quad \ldots \right\}$*, where*

for any $\bar{x} \in m^i_j \ominus \boldsymbol{s}^i_{j_2^*}$*,* $\bar{x} \in \mathcal{W}_{j_2^*}$ *if* $\boldsymbol{s}^i_{j_2^*} \oplus \{\bar{x}\}$ *is exclusive in* m^i_j*;*

(b) $F_{j_2^*} = \left\{ d_1^{F_{j_2^*}} \quad d_2^{F_{j_2^*}} \quad \ldots \right\}$*, where*

for any $d' \in F$*,* $d' \in F_{j_2^*}$ *if* $d(\boldsymbol{s}^i_{j_2^*}) \oplus \{d'\}$ *is exclusive in* F*.*

Since $d(s^i_\iota)$ *is not found, we continue to Step 2 with* $\bar{x}_{p_1^*}^{\mathcal{W}_{j_1^*}}$*,* $d(\bar{x}_{p_1^*}^{\mathcal{W}_{j_1^*}})$*,* $\mathcal{W}_{j_2^*}$ *and* $F_{j_2^*}$*.*

Step 2: The most suitable mapping $d(\bar{x}_{p_2^*}^{\mathcal{W}_{j_2^*}})$ *is derived for* $\bar{x}_{p_2^*}^{\mathcal{W}_{j_2^*}} \in F_{j_2^*}$ *in*

(3.20).

$$\begin{aligned}\bar{x}_{p_2^*}^{\mathcal{W}_{j_2^*}} &= \arg\min_{\bar{x}\in\mathcal{W}_{j_2^*}}\ \min_{d(\bar{x})\in F_{j_2^*}} L_{\boldsymbol{s}^i_{j_2^*}\oplus\{\bar{x}\}},\\ d(\bar{x}_{p_2^*}^{\mathcal{W}_{j_2^*}}) &= \arg\min_{d(\bar{x}_{p_2^*}^{\mathcal{W}_{j_2^*}})\in F_{j_2^*}} L_{\boldsymbol{s}^i_{j_2^*}\oplus\left\{\bar{x}_{p_2^*}^{\mathcal{W}_{j_2^*}}\right\}}.\end{aligned} \tag{3.20}$$

We have $\boldsymbol{s}^i_{j_3^*} = \boldsymbol{s}^i_{j_2^*} \oplus \left\{\bar{x}_{p_2^*}^{\mathcal{W}_{j_2^*}}\right\}$.
If $s^i_\iota \subset \boldsymbol{s}^i_{j_3^*}$*, then* $d(s^i_\iota) \subset d(\boldsymbol{s}^i_{j_3^*})$ *and continue to Step 0 of the next iteration.*
If $s^i_\iota \not\subset \boldsymbol{s}^i_{j_3^*}$*, then*

(a) $\mathcal{W}_{j_3^*} = \left\{\ \bar{x}_1^{\mathcal{W}_{j_3^*}} \quad \bar{x}_2^{\mathcal{W}_{j_3^*}} \quad \ldots\ \right\}$*, where*

for any $\bar{x} \in m^i_j \ominus \boldsymbol{s}^i_{j_3^*}$*,* $\bar{x} \in \mathcal{W}_{j_3^*}$ *if* $\boldsymbol{s}^i_{j_3^*} \oplus \{\bar{x}\}$ *is exclusive in* m^i_j*;*

(b) $F_{j_3^*} = \left\{\ d_1^{F_{j_3^*}} \quad d_2^{F_{j_3^*}} \quad \ldots\ \right\}$*, where*

for any $d' \in F$*,* $d' \in F_{j_3^*}$ *if* $d(\boldsymbol{s}^i_{j_3^*}) \oplus \{d'\}$ *is exclusive in* F*.*

Since $d(s^i_\iota)$ *is not found, we continue to Step 3 with* $\bar{x}_{p_2^*}^{\mathcal{W}_{j_2^*}}$*,* $d(\bar{x}_{p_2^*}^{\mathcal{W}_{j_2^*}})$*,* $\mathcal{W}_{j_3^*}$ *and* $F_{j_3^*}$*.*

$\vdots$

Step (q-1): The most suitable mapping $d(\bar{x}_{p_{q-1}^*}^{\mathcal{W}_{j_{q-1}^*}})$ *is derived for* $\bar{x}_{p_{q-1}^*}^{\mathcal{W}_{j_{q-1}^*}} \in F_{j_{q-1}^*}$ *in (3.21).*

$$\begin{aligned}\bar{x}_{p_{q-1}^*}^{\mathcal{W}_{j_{q-1}^*}} &= \arg\min_{\bar{x}\in\mathcal{W}_{j_{q-1}^*}}\ \min_{d(\bar{x})\in F_{j_{q-1}^*}} L_{\boldsymbol{s}^i_{j_{q-1}^*}\oplus\{\bar{x}\}},\\ d(\bar{x}_{p_{q-1}^*}^{\mathcal{W}_{j_{q-1}^*}}) &= \arg\min_{d(\bar{x}_{p_{q-1}^*}^{\mathcal{W}_{j_{q-1}^*}})\in F_{j_q^*}} L_{\boldsymbol{s}^i_{j_q^*}\oplus\left\{\bar{x}_{p_{q-1}^*}^{\mathcal{W}_{j_{q-1}^*}}\right\}}.\end{aligned} \tag{3.21}$$

We have $\boldsymbol{s}^i_{j_q^*} = \boldsymbol{s}^i_{j_{q-1}^*} \oplus \left\{\bar{x}_{p_{q-1}^*}^{\mathcal{W}_{j_{q-1}^*}}\right\}$.
If $s^i_\iota \subset \boldsymbol{s}^i_{j_q^*}$*, then* $d(s^i_\iota) \subset d(\boldsymbol{s}^i_{j_q^*})$ *and continue to Step 0 of the next iteration.*
If $s^i_\iota \not\subset \boldsymbol{s}^i_{j_q^*}$*, then*

(a) $\mathcal{W}_{j_q^*} = \left\{\ \bar{x}_1^{\mathcal{W}_{j_q^*}} \quad \bar{x}_2^{\mathcal{W}_{j_q^*}} \quad \ldots\ \right\}$*, where*

for any $\bar{x} \in m_j^i \ominus \boldsymbol{s}_{j_q^*}^i$, $\bar{x} \in \mathcal{W}_{j_q^*}$ *if* $\boldsymbol{s}_{j_q^*}^i \oplus \{\bar{x}\}$ *is exclusive in* m_j^i*;*

(b) $F_{j_q^*} = \left\{ \begin{array}{ccc} d_1^{F_{j_q^*}} & d_2^{F_{j_q^*}} & \dots \end{array} \right\}$*, where*

for any $d' \in F$, $d' \in F_{j_q^*}$ *if* $d(\boldsymbol{s}_{j_q^*}^i) \oplus \{d'\}$ *is exclusive in* F*.*

Since $d(s_\iota^i)$ *is not found, we continue to Step q with* $\bar{x}_{p_{q-1}^*}^{\mathcal{W}_{j_{q-1}^*}}$, $d(\bar{x}_{p_{q-1}^*}^{\mathcal{W}_{j_{q-1}^*}})$, $\mathcal{W}_{j_q^*}$ *and* $F_{j_q^*}$*.*

Step q: The most suitable mapping $d(\bar{x}_{p_q^*}^{\mathcal{W}_{j_q^*}})$ *is derived for* $\bar{x}_{p_q^*}^{\mathcal{W}_{j_q^*}} \in F_{j_q^*}$ *in (3.22).*

$$\begin{aligned} \bar{x}_{p_q^*}^{\mathcal{W}_{j_q^*}} &= \arg \min_{\bar{x} \in \mathcal{W}_q} \min_{d(\bar{x}) \in F_{j_q^*}} L_{\boldsymbol{s}_{j_q^*}^i \oplus \{\bar{x}\}}, \\ d(\bar{x}_{p_q^*}^{\mathcal{W}_{j_q^*}}) &= \arg \min_{d(\bar{x}_{p_q^*}^{\mathcal{W}_{j_q^*}}) \in F_{j_{q+1}^*}} L_{\boldsymbol{s}_{j_{q+1}^*}^i \oplus \left\{\bar{x}_{p_q^*}^{\mathcal{W}_{j_q^*}}\right\}}. \end{aligned} \tag{3.22}$$

We have $\boldsymbol{s}_{j_{q+1}^*}^i = \boldsymbol{s}_{j_q^*}^i \oplus \left\{\bar{x}_{p_q^*}^{\mathcal{W}_{j_q^*}}\right\}$.
If $s_\iota^i \subset \boldsymbol{s}_{j_{q+1}^*}^i$*, then* $d(s_\iota^i) \subset d(\boldsymbol{s}_{j_{q+1}^*}^i)$ *and continue to Step 0 of the next iteration.*
If $s_\iota^i \not\subset \boldsymbol{s}_{j_{q+1}^*}^i$*, then*

(a) $\mathcal{W}_{j_{q+1}^*} = \left\{ \begin{array}{ccc} \bar{x}_1^{\mathcal{W}_{j_{q+1}^*}} & \bar{x}_2^{\mathcal{W}_{j_{q+1}^*}} & \dots \end{array} \right\}$*, where*

for any $\bar{x} \in m_j^i \ominus \boldsymbol{s}_{j_{q+1}^*}^i$, $\bar{x} \in \mathcal{W}_{j_{q+1}^*}$ *if* $\boldsymbol{s}_{j_{q+1}^*}^i \oplus \{\bar{x}\}$ *is exclusive in* m_j^i*;*

(b) $F_{j_{q+1}^*} = \left\{ \begin{array}{ccc} d_1^{F_{j_{q+1}^*}} & d_2^{F_{j_{q+1}^*}} & \dots \end{array} \right\}$*, where*

for any $d' \in F$, $d' \in F_{j_{q+1}^*}$ *if* $d(\boldsymbol{s}_{j_{q+1}^*}^i) \oplus \{d'\}$ *is exclusive in* F*.*

Since $d(s_\iota^i)$ *is not found, we continue to Step (q+1) with* $\bar{x}_{p_q^*}^{\mathcal{W}_{j_q^*}}$, $d(\bar{x}_{p_q^*}^{\mathcal{W}_{j_q^*}})$, $\mathcal{W}_{j_{q+1}^*}$ *and* $F_{j_{q+1}^*}$*.*

The process continues until $d(s_\iota^i)$ *is found. Then, return to Step 0 of the next iteration.*

(B) If there are two robots in $s_\iota^i = \left\{x_1^{s_\iota^i}, x_2^{s_\iota^i}\right\} \in s^i$*, then robot* x_i *could follow*

(3.23) below to solve $d(s_\iota^i)$ for s_ι^i:

$$d(s_\iota^i) = \arg \min_{d(s_\iota^i)\in\Pi} L_{s_\iota^i}. \tag{3.23}$$

Note that through Protocol 3.5.1, we find $d(s_\iota^i)$ for any s_ι^i derived from g_i. All the s_ι^i includes all the robots in g_i and are sufficient to produce the control force in (3.8). Unanimous mapping on $d(s_\iota^i)$ is obtained for the robot groups in w_j^i, whose communal group reference is m_j^i. However, disagreements are still possible among robots that can not access the same communal group reference. The control law drives these robots to a partially desired formation, where these robots can access the same communal group reference under the displacement-based reference formation. Then, Protocol 3.5.1 triggers a change in the mapping to avoid the mapping collisions. The avoidance is possible upon connectivity of the robot formation and the displacement framework in defining the reference formation. It takes seconds to update the loop in Matlab simulation. In practice, the proposed algorithms will be optimised for hardware such that system response can be real-time.

Thus, we still need to establish the connectivity condition. Suppose that the robot formation remains connected from instant k_0 to k. As the robot formation is connected, there may be multiple paths between two neighbouring robots x_i and x_j. From these paths, there exists a path-dependent $V_{ij}^*(k)$ such that

$$\begin{aligned} x_i - x_j &= V_{ij}(k)\hat{\delta}(k), \\ V_{ij}^*(k) &= \arg \min_{V_{ij}(k)} |V_{ij}(k)\hat{\delta}(k)|. \end{aligned}$$

Given that $x_i \in s_\iota^i \in s^i$ and $x_j \in s_{\iota'}^j \in s^j$, we have

$$\begin{aligned} |x_i(k) - x_j(k)| &\le |x_i(k) - x_j(k) - d^{s_\iota^i}(x_i(k)) + d^{s_{\iota'}^j}(x_j(k))| \\ &\quad +|d^{s_\iota^i}(x_i(k)) - d^{s_{\iota'}^j}(x_j(k))| \\ &= |V_{ij}^*(k)\hat{\delta}(k)| + |d^{s_\iota^i}(x_i(k)) - d^{s_{\iota'}^j}(x_j(k))|. \end{aligned}$$

Assume that $|V_{ij}^*(k)\hat{\delta}(k)| + |d^{s_\iota^i}(x_i(k)) - d^{s_{\iota'}^j}(x_j(k))|$ that is dependent on the control edges from a particular path decreases under the control forces. Then, to maintain connectivity of the robot formation, we have

$$|x_i(k) - x_j(k)| \le |V_{ij}^*(k_0)\hat{\delta}(k_0)| + |d^{s_\iota^i}(x_i(k_0)) - d^{s_{\iota'}^j}(x_j(k_0))| \le R.$$

To sum up, the condition is given in (3.24). When $k_0 = 0$ at the initial instant, the robot formation is always connected.

$$\begin{aligned} &R \geq |\hat{\delta}(k_0)| \max |V_{ij}^*(k_0)| + \max |d^{s_\iota^i}(x_i(k_0)) - d^{s_{\iota'}^j}(x_j(k_0))|, \;\; \textit{or} \\ &R \geq |\hat{\delta}(k_0)| \max |V_{ij}^*(k_0)| + \max_{\delta d \in \Upsilon} |\delta d|. \end{aligned} \tag{3.24}$$

Note that the condition in (3.24) implies a priori knowledge of the initial states. We can relax this condition. For three robots to be communal neighbours, the minimum communication radius R is the circumcircle radius of the triangle from them. Moreover, it can be proved that the perimeter of a triangle is no bigger than $3\sqrt{3}$ times its circumcircle radius. According to (3.6), the circumcircle radius of the triangle of a three-edge control group is at most R. Thus, the perimeter is at most $3\sqrt{3}R$. We have

$$\sqrt{3}R \sum_{i=1}^{n} \rho_i(k) \geq \sum_{\hat{\delta}_j(k) \in \hat{\delta}(k)} |\hat{\delta}_j(k)| \geq 2|\hat{\delta}(k_0)| \min |V_{ij}^*(k_0)| \sum_{i=1}^{n} \rho_i(k).$$

where robot x_i is involved in $\rho_i(k)$ control groups instant k. Assume that $\sigma \geq \max |V_{ij}^*(k_0)| - \min |V_{ij}^*(k_0)|$ is dependent on the robot formation topology. We have

$$\begin{aligned} &\sqrt{3}R \sum_{i=1}^{n} \rho_i(k) \geq 2|\hat{\delta}(k_0)|(\max |V_{ij}^*(k_0)| - \sigma) \sum_{i=1}^{n} \rho_i(k), \\ &\frac{\sqrt{3}R}{2|\hat{\delta}(k_0)|} + \sigma \geq \max |V_{ij}^*(k_0)|. \end{aligned}$$

By (3.24), we have

$$\begin{aligned} R &\geq \frac{R\sqrt{3}}{2} + \sigma|\hat{\delta}(k_0)| + \max_{\delta d \in \Upsilon} |\delta d|, \\ R &\geq \frac{2(\sigma|\hat{\delta}(k_0)| + \max_{\delta d \in \Upsilon} |\delta d|)}{2 - \sqrt{3}}. \end{aligned} \tag{3.25}$$

Thus, if (3.24) or (3.25) is satisfied, then the formation remains connected. However, it can be seen that such connectivity condition requires global information, which is unobtainable to the robots. Therefore, the deduction above is only for the sake of analysis.

3.6 Simulation Examples

In this section, simulation results are given to demonstrate the effectiveness of the proposed protocol. A first-order robot formation with 9 robots is used, and the displacement-based reference formation $\Delta\Upsilon$ is given below.

$$\Delta\Upsilon = \left\{ \begin{array}{cccc} \Delta_1\Upsilon & \Delta_2\Upsilon & \dots & \Delta_8\Upsilon \end{array} \right\}, \tag{3.26}$$

where

$$\begin{aligned}
\Delta_1\Upsilon &= \left\{ \begin{bmatrix} -6 \\ -18 \end{bmatrix} \begin{bmatrix} -6 \\ 9 \end{bmatrix} \begin{bmatrix} 12 \\ 9 \end{bmatrix} \right\}, \\
\Delta_2\Upsilon &= \left\{ \begin{bmatrix} -12 \\ -9 \end{bmatrix} \begin{bmatrix} -6 \\ 9 \end{bmatrix} \begin{bmatrix} 18 \\ 0 \end{bmatrix} \right\}, \\
\Delta_3\Upsilon &= \left\{ \begin{bmatrix} 6 \\ -9 \end{bmatrix} \begin{bmatrix} -18 \\ 0 \end{bmatrix} \begin{bmatrix} 12 \\ 9 \end{bmatrix} \right\}, \\
\Delta_4\Upsilon &= \left\{ \begin{bmatrix} -12 \\ -9 \end{bmatrix} \begin{bmatrix} 6 \\ 18 \end{bmatrix} \begin{bmatrix} 6 \\ -9 \end{bmatrix} \right\}, \\
\Delta_5\Upsilon &= \left\{ \begin{bmatrix} 12 \\ 9 \end{bmatrix} \begin{bmatrix} -12 \\ -9 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} \right\}, \\
\Delta_6\Upsilon &= \left\{ \begin{bmatrix} 18 \\ 0 \end{bmatrix} \begin{bmatrix} -18 \\ 0 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} \right\}, \\
\Delta_7\Upsilon &= \left\{ \begin{bmatrix} 6 \\ 18 \end{bmatrix} \begin{bmatrix} -6 \\ -18 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} \right\}, \\
\Delta_8\Upsilon &= \left\{ \begin{bmatrix} 6 \\ -9 \end{bmatrix} \begin{bmatrix} -6 \\ 9 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \end{bmatrix} \right\}.
\end{aligned}$$

Note that $\Delta\Upsilon$ given in (3.26) satisfies its definition in (3.6). Each robot uses $\Delta\Upsilon$ and the displacements from the sensors as the reference for the protocol to produce mapping decisions. Based on the mapping decision and the displacements from the sensors, its controller then generates a momentum.

We assume that the robots are equipped with short-range sensors with a radius $R = 21$m. The robot formation is connected at zero instant t_0. We denote t_s as the instant when the average square root of control edges $\frac{\sqrt{\hat{\delta}^T\hat{\delta}}}{\sum_i^n \varkappa_i(k)}$ is smaller than 0.1m, where $n = 9$ and $\varkappa_i(k)$ denotes the number of robots in g_i for robot x_i at instant t_k. The simulation ceases at this time.

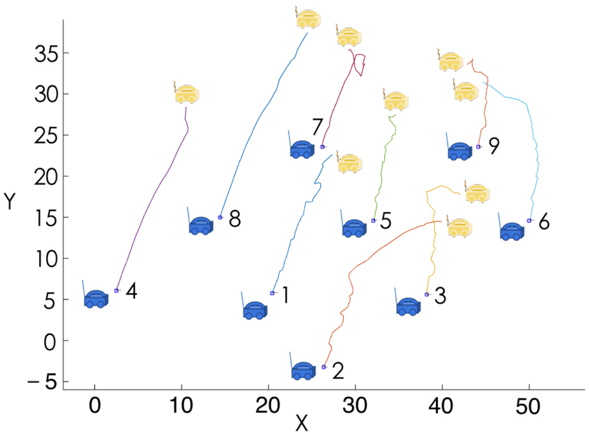


Figure 3.1: Trajectories of robots, with the yellow and the blue cars as the initial and the final positions of the robots respectively.

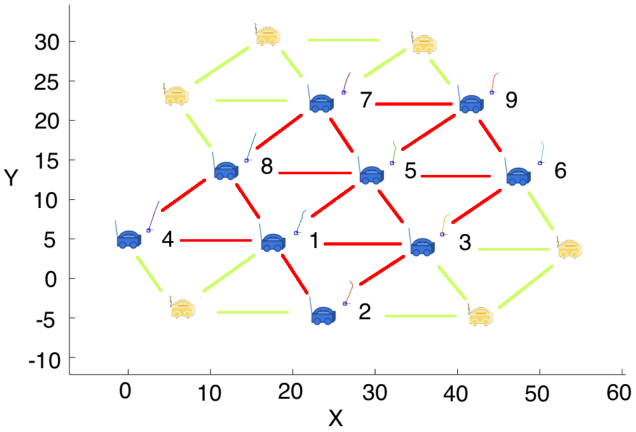
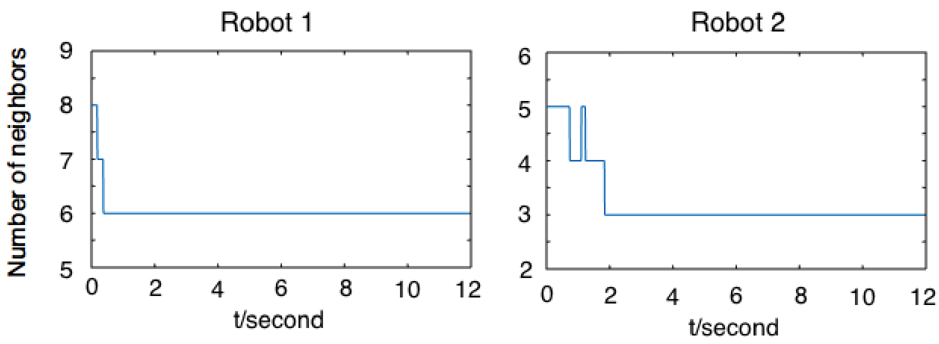


Figure 3.2: Formation when $\frac{\sqrt{\hat{\delta}^T\hat{\delta}}}{\sum_i^n \varkappa_i(k)}$ is smaller than 0.1m, with the blue and the yellow cars as the real positions and some of the extendible positions.



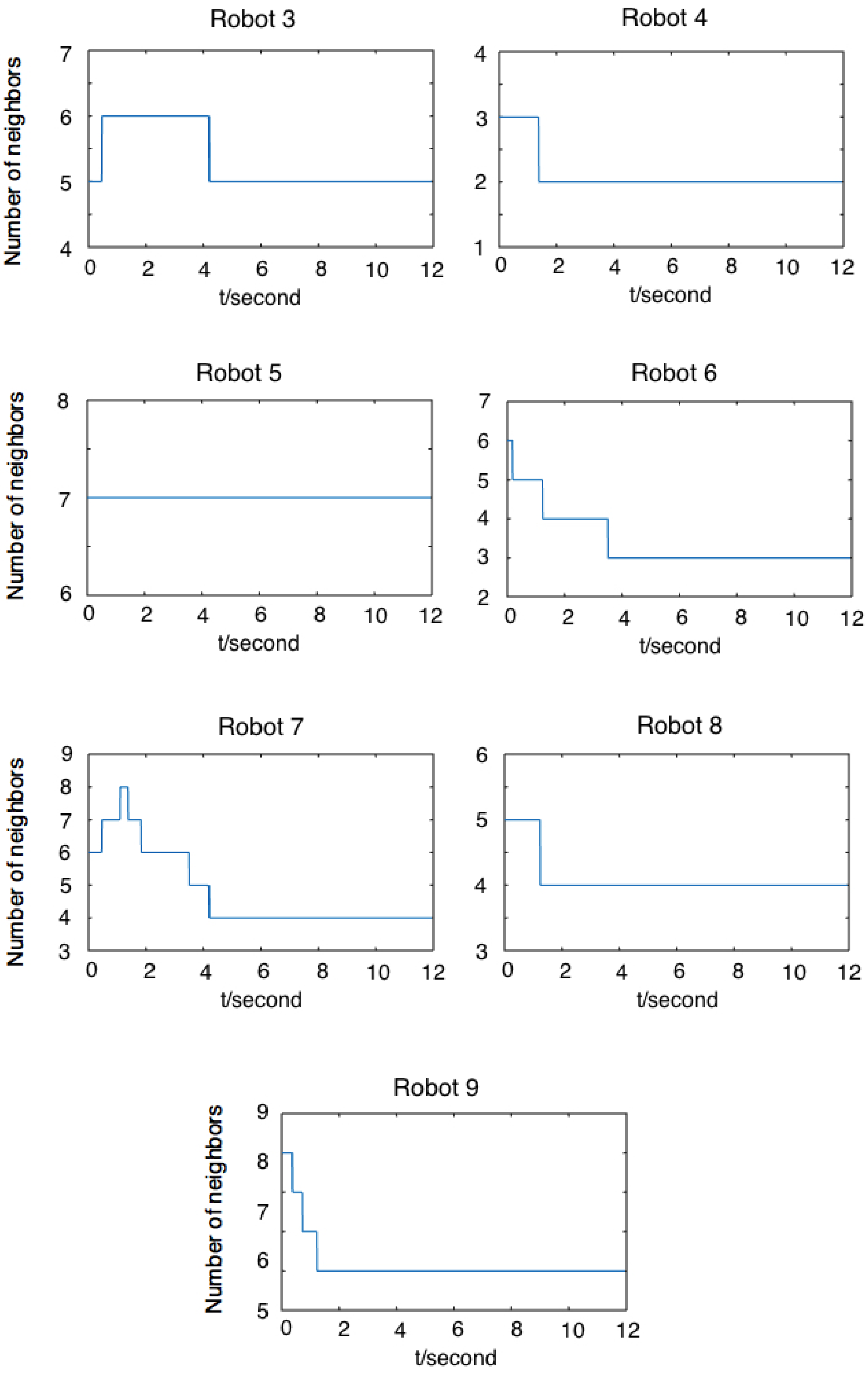


Figure 3.3: Time and number of neighbours robots for each robot.

Table 3.1: The groups of neighbours in the final formation $x(t_s)$ using Protocol 1.

t_s/second	$\{neighbours\ of\ robot\ x_i\}_i$
8.47	$\{2,4\}_1, \{1\}_2, \{6\}_3, \{1,7\}_4, \{6\}_5, \{3,5,8\}_6, \{4\}_7, \{6,9\}_8, \{8\}_9$
7.15	$\{2\}_1, \{1,4,6\}_2, \{6\}_3, \{2,5\}_4, \{4,6,8,9\}_5, \{2,3,5\}_6, \{8\}_7, \{5,7\}_8, \{5\}_9$
6.98	$\{5\}_1, \{5\}_2, \{6\}_3, \{5,7,8\}_4, \{1,2,4\}_5, \{3,9\}_6, \{4\}_7, \{4,9\}_8, \{6,8\}_9$
4.43	$\{2,4,5\}_1, \{1,3\}_2, \{2,5,6\}_3, \{1,7\}_4, \{1,3,7,9\}_5, \{3,9\}_6, \{4,5,8\}_7, \{7,9\}_8, \{5,6,8\}_9$
6.03	$\{4\}_1, \{5,6\}_2, \{6\}_3, \{1,5\}_4, \{2,4,7,9\}_5, \{2,3,9\}_6, \{5,8,9\}_7, \{7,9\}_8, \{5,6,7,8\}_9$
6.25	$\{4\}_1, \{4,5\}_2, \{5\}_3, \{1,2,7,8\}_4, \{2,3,6,8\}_5, \{5,8,9\}_6, \{4\}_7, \{4,5,6,9\}_8, \{6,8\}_9$
9.34	$\{4\}_1, \{5,6\}_2, \{6,9\}_3, \{1,5,7\}_4, \{2,4,8\}_5, \{2,3,8\}_6, \{4,8\}_7, \{5,6,7,9\}_8, \{3,8\}_9$
6.53	$\{2,4,5\}_1, \{1,3\}_2, \{2,5\}_3, \{1,5,7,8\}_4, \{1,3,4,8\}_5, \{9\}_6, \{4\}_7, \{4,5\}_8, \{6\}_9$
4.65	$\{2\}_1, \{1,4,5\}_2, \{5,6\}_3, \{2,7,8\}_4, \{2,3,6,8,9\}_5, \{3,5,9\}_6, \{4\}_7, \{4,5\}_8, \{5,6\}_9$

Note that Figures 3.1 and 3.2 are trajectories of the robots and their final formation, respectively, in both of which the identity of each robot has been labeled. Figure 3.1 shows that robots are able to adjust their movements according to their neighbours such that each of them finds its desired position in the final formation. In Figure 3.2, it can be seen from the parallel trajectories that as the misalignments are approaching zero, the speed consensus of the robots is also reached. In Figure 3.3, it can be seen that even though the numbers of robots' neighbours change, the robots are able to interact and reach a desired formation. Compared with those in the initial formations, robots tend to have fewer neighbours in the final states. For example, robots 6 and 9 have most neighbours at instant zero. However, judging from the positions, they should move to the edge and have much less neighbours. Thus, as the protocol takes place, their neighbours become less. This suggests that under the protocol, the robots scatter and are reasonably relocated to reach a desired formation.

With different initial formations $x(t_0)$, we have different final formations

$x(t_s)$. From these different final formations, individual robots have different groups of neighbours in different rows of Table 3.1. For example in the second row of the second column of Table 3.1, from the final formation $x(t_s = 8.47\,\mathrm{s})$ of the 9 robots, robot x_1's neighbours are robots x_2 and x_4, which is represented by $\{2,4\}_1$; robot x_2's neighbor is robot x_1 by $\{1\}_2$; robot x_6 is robot x_3's neighbor by $\{6\}_3$; robot x_4's neighbours are robots x_1 and x_7 by $\{1,7\}_4$; robot x_6 is robot x_5's neighbor by $\{6\}_5$; robot x_6's neighbours are robots x_3, x_5 and x_8 by $\{3,5,8\}_6$; robot x_7's neighbor is robot x_4 by $\{4\}_7$; robot x_8's neighbours are robots x_6 and x_9 by $\{6,9\}_8$; robot x_9's neighbor is robot x_8 by $\{8\}_9$. Moreover, the differences among these different initial formations $x(t_0)$ are subject to limited randomness. It can be seen from Table 3.1 that some robots are more likely to be neighbours than others, for example, robots 4 and 7, robots 5 and 8, and robots 8 and 9. The differences among different initial formations leads to differences among their final formations. This suggests that the robot team has some level of self-organising under Protocol 1.

3.7 Summary

In order to make the robot team size scalable, the concept of displacement-based formation based on communal neighbours is proposed. Data transmission is not available and thus the novel controller that drives the robots to a partially desired formation is used. The distributed coordination protocol is then devised based on communal neighbours, communal group reference, and the concept of exclusiveness. The protocol is able to map a group of communal neighbours to reach unanimous mappings, and under a given condition on the sensing range, connectivity is maintained. Future work includes increasing the formation complexity, introducing polygon to the coordination protocol, and improving the mapping algorithm, as computation burden is heavier with increased complexity of the formation and the protocol.

Chapter 4

Arbitrary Formation with No Communication

Formation control for a team of robots is investigated in this chapter. We consider the case where no communication channels are available and only sensors with limited ranges can provide the robots with the relative displacements of nearby robots. The robots themselves are fully independent with no designated roles. Thus, the robot team is distributed both physically and in their information flow. To deal with the lack of communication, a novel controller is designed to drive the robots to a partially desired formation. Then, a coordination protocol is proposed to detect the imperfections and transform the locally desired formations into a globally desired formation. To avoid local optimum traps, an evaluation method imposes a weak restriction on the predefined formation, rendering it almost arbitrary. To enhance the robustness, the minimal local topology is proposed, and to reduce the computation burden and avoid the infinite trajectory loop, the coordination protocol is modified by introducing probability. Simulations are conducted to verify the performance.

4.1 Introduction

In this chapter, we investigate the formation control problem for a team of robots. The robots are equipped with displacement sensors of limited ranges, see for example, [40, 43, 46, 47, 49]. Data communication among them is not available. This represents a more difficult problem than those of [40, 43, 44, 46, 49], which reserve at least limited communication or some global information. Thus, the robot formation has time-switching formation topologies rather than fixed formations in [41, 42, 46]. To deal with the lack of access to data transmission, a novel controller is used rather than the artificial potential field in [45]. The control law can drive the robots to a potentially flawed formation where the imperfections can be detected by the coordination protocol. Instead of assuming hard restrictions on formation shape [42, 47, 49], the proposed strategy is applicable to almost arbitrary formations. To enhance system robustness and reduce conservativeness on maintaining system connectivity, the minimal local topology is proposed. To reduce computation burden and avoid the trap of infinite loops, the protocol is modified by introducing probability.

Notation. The notation used is standard. The superscript "T" stands for matrix transposition; the notation $|\cdot|$ denotes the length of a vector; $P > 0$ means that P is real positive definite; $\mathbf{R}^m$ is an m-th dimensional space of real numbers. For operations defined on groups $A = \{a_1, a_2, \dots\}$, $B = \{b_1, b_2, \dots\}$, we have $C = A \oplus B = \{c_1, c_2, \dots\}$, where $c \in A$ or $c \in B$ for any $c \in C$; $C = A \ominus B = \{c_1, c_2, \dots\}$, where $c \in A$ and $c \notin B$ for any $c \in C$; $C = A \odot B = \{c_1, c_2, \dots\}$, where $c \in A$ and $c \in B$ for any $c \in C$; operations defined on matrices follow the common definitions. Denote $a \in A$ if a member a belongs to a set A; $B \subseteq A$ if B is a subset of A by $B \ominus A = \emptyset$. The bold and the non-bold styles of a symbol denote different variables; the italic and the non-italic styles of a symbol denote the same variable. Matrices are assumed to be compatible for algebraic operations.

4.2 Problem Formulation

This chapter investigates the formation control problem for a team of autonomous robots. The robots are mass-produced from the same stencil and they are equally functional. None of the robots has designated roles as leaders or followers, and the formation positions are interchangeable among the robots. Therefore, they need to map themselves to the positions in the predefined formation by a sequence of individual decisions. Each robot is equipped with a displacement sensor that has only a limited sensing range because of its limited available power and filtering complexity. In a typical surveillance application, the navigation is required to be in stealth mode. Active radio communication is undesired because the signal is often weak, vulnerable and unreliable in a complex environment. Because of the limited sensing range and the lack of data communication, the information flow cannot be global and must be localised when performing the team formation.

As described above, the displacement sensors can provide the robots with local measurements of the relative positions from their in-range neighbours. Thus, though subject to localisation, the robots have access to both local measurements and the predefined formation. The objective of the robot team is to reach a desired formation from the predefined formation. The challenge confronted by the team is that none of the robots has any designated position; they can only identify and track other robots within the range but they cannot differentiate them and once some robots move out of the range, they are immediately forgotten. On account of their random initial positions, the robots need to make their own mapping decisions based on local measurements and the predefined formation, and try to reduce the wandering time and trajectory as much as possible.

4.3 Design of Control Law and Coordination Protocol

In this section, we first derive what is available under the limitation of sensing range and unavailable communication. Then, based on this local information, we propose our controller and protocol design.

4.3.1 System Description

The objective formation is a predefined point group

$$\mathbf{D} = \left\{ \begin{array}{cccc} d_1 & d_2 & \dots & d_N \end{array} \right\} \text{ in set form, or } D = \begin{bmatrix} d_1 \\ d_2 \\ \vdots \\ d_N \end{bmatrix} \text{ in matrix form}$$

in space $\mathbf{R}^2$, where $d' \in \mathbf{R}^2$ for any $d' \in \mathbf{D}$. The objective is to drive a robot team

$$\mathbf{X} = \left\{ \begin{array}{cccc} x_1 & x_2 & \dots & x_N \end{array} \right\} \text{ in set form, or } X = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix} \text{ in matrix form}$$

of N mobile robots in space $\mathbf{R}^2$ to the relative displacements from the predefined formation $\mathbf{D}$, where $x' \in \mathbf{R}^2$ for any $x \in \mathbf{X}$. This means that if the predefined formation is reached, then for any $d', d'' \in \mathbf{D}$, there is $x', x'' \in \mathbf{X}$ that $x' - x'' = d' - d''$.

No robot has any designated position and the robots are fully independent. These autonomous robots need to make mapping decisions individually and are able to behave according to a control law. The problem is how each robot can make a series of temporal sequential decisions so that ultimately, a one-to-one mapping can be achieved from the robot team $\mathbf{X}$ to the predefined formation $\mathbf{D}$.

For a robot team communicating in a distributed manner, particularly with multiple overlapped robot communities, agreements in each community can be used to eliminate discrepancies among interweaving communities. Thus, by local cooperation, a global consensus is possible. Similarly, as overlapped areas of sensing ranges could be used to establish communities, agreements from these local communities can help the robot team reach a global consensus.

We now present a framework of deriving useful sets from available local information for constructing control force and coordination protocol. For robot x_i under the sensing range R, we can define its neighbourhood $\mathcal{N}(\mathbf{X}, R, x_i)$ as below.

$$\mathcal{N}(\mathbf{X}, R, x_i) = \left\{ \begin{array}{ccc} x_1^i & x_2^i & \dots \end{array} \right\}, \tag{4.1}$$

where for any $x' \in \mathbf{X}$, $x' \in \mathcal{N}(\mathbf{X}, R, x_i)$ if $|x' - x_i| \leq R$. Based on (4.1) and under the sensing range R, we can define a group G^i of exclusive mutual neighbor groups G_k^i, $1 \leq k \leq n(G^i)$, around robot x_i. Note that for a set

$Y = \left\{ \begin{array}{ccc} Y_1 & \dots & Y_{n(Y)} \end{array} \right\}$, its degree is $n(Y)$.

$$
\begin{aligned}
G^i_k &= \left\{ \begin{array}{ccc} x^{i,k}_1 & x^{i,k}_2 & \dots \end{array} \right\}, \\
G^i &= \left\{ \begin{array}{ccc} G^i_1 & G^i_2 & \dots \end{array} \right\}, \\
G &= \left\{ \begin{array}{ccc} G^1 & \dots & G^N \end{array} \right\},
\end{aligned}
\tag{4.2}
$$

where i) $|x' - x''| \leq R$ for any $G^i_k \in G^i$ and any $x', x'' \in G^i_k$, and ii) for any $x''' \in \mathbf{X}$, $x''' \in G^i_k$ if there is $\bar{x}' \in \mathcal{N}(\mathbf{X}, R, x_i) \ominus G^i_k$ that $|\bar{x}' - x'''| > R$. Note that G^i_k implies global coordinates. It can, however, be seen in Sections 4.3.2 and 4.3.3 that only $\bar{G}^i_k$ below is used to produce any mapping and control.

$$
\bar{G}^i_k = \left\{ \begin{array}{ccc} x^{i,k}_1 - x_i & x^{i,k}_2 - x_i & \dots \end{array} \right\},
\tag{4.3}
$$

where $x' - x_i \in \bar{G}^i_k$ for any $x' \in G^i_k$. As the displacement sensors are able to provide robot x_i with $\bar{G}^i_k$, the need of global information is avoided.

Note that G^i is derived by robot x_i, who can only access $\mathcal{N}(\mathbf{X}, R, x_i)$ because of limited sensing range. All the elements in G^i are from $\mathcal{N}(\mathbf{X}, R, x_i)$ and all the elements in $\mathcal{N}(\mathbf{X}, R, x_i)$ are included in G^i, as is shown in (4.4). Moreover, if we gather values from $\mathbf{X}$ and produce a new set G'^i to include all the exclusive mutual neighbor groups containing robot x_i, then it still follows that $G'^i = G^i$. Similarly, all the elements in G are from $\mathbf{X}$ and all the elements in $\mathbf{X}$ are included in G, as is shown in (4.5). Such a representation of G^i makes sure that we take full advantage of $\mathcal{N}(\mathbf{X}, R, x_i)$.

$$
\mathcal{N}(\mathbf{X}, R, x_i) = \left\{ \begin{array}{ccc} x'_1 & x'_2 & \dots \end{array} \right\},
\tag{4.4}
$$

where for any $x'_l \in \mathcal{N}(\mathbf{X}, R, x_i)$, there exists $G^i_k \in G^i$ that $x'_l \in G^i_k$.

$$
\mathbf{X} = \left\{ \begin{array}{ccc} x''_1 & x''_2 & \dots \end{array} \right\},
\tag{4.5}
$$

where for any $x''_{l'} \in \mathbf{X}$, there exists $G^i_k \in G^i \in G$ that $x''_{l'} \in G^i_k \in G^i$.

We assume that neighbouring robots should be mapped to neighbouring points in the predefined formation. Since neighbouring robots are defined in (4.2), neighbouring points in $\mathbf{D}$ also need to be defined. Similarly with (4.2), we divide $\mathbf{D}$ into a set $U\mathbf{D}$ of groups $U^i\mathbf{D}$, $1 \leq i \leq n(\mathbf{D})$, of exclusive mutual neighbor groups $U^i_k\mathbf{D}$, $1 \leq k \leq n(U^i\mathbf{D})$. Here, U can be seen as a division operator on set $\mathbf{D}$.

$$
\begin{aligned}
U\mathbf{D} &= \left\{\; U^1\mathbf{D} \;\; \dots \;\; U^N\mathbf{D} \;\right\}, \;\; U^i\mathbf{D} = \left\{\; U_1^i\mathbf{D} \;\; U_2^i\mathbf{D} \;\; \dots \;\right\}, \\
U_k^i\mathbf{D} &= \left\{\; d_1^{i,k} \;\; d_2^{i,k} \;\; \dots \;\right\}, \\
UD &= \begin{bmatrix} U^1D \\ \vdots \\ U^ND \end{bmatrix}, \;\; U^iD = \begin{bmatrix} U_1^iD \\ U_2^iD \\ \vdots \end{bmatrix}, \;\; U_k^iD = \begin{bmatrix} d_1^{i,k} \\ d_2^{i,k} \\ \vdots \end{bmatrix},
\end{aligned}
\tag{4.6}
$$

where i) $n(U_k^i\mathbf{D}) = i$ for $1 \leq i \leq n(\mathbf{D})$, and ii) for any $d_p^{i,k}, d_q^{i,k} \in U_k^i\mathbf{D}$, there is $d_{p'}^{i,k} \in U_k^i\mathbf{D}$ such that $|d_p^{i,k} - d_q^{i,k}| < |\bar{d}_{q'}^{i,k} - d_{p'}^{i,k}|$ for any $\bar{d}_{q'}^{i,k} \in \mathbf{D} \ominus U_k^i\mathbf{D}$.

The importance of $U\mathbf{D}$ is to provide a mapping source for any G_k^i. Therefore, to ensure that any G_k^i can find its mapping counterpart $U_{k'}^{i'}\mathbf{D}$ with $n(G_k^i) = n(U_{k'}^{i'}\mathbf{D})$, the following assumption is made.

Assumption 4.3.1. *For any $d_i \in \boldsymbol{D}$, there are no $d_j, d_k \in \boldsymbol{D}$ such that $|d_i - d_j| = |d_i - d_k|$.*

Now we have defined G^i and $U^i\mathbf{D}$, thus enabling mapping $d(G_i) = \mathcal{P}(G^i, U\mathbf{D})$ from neighbouring robots G^i to neighbouring points $U\mathbf{D}$, where G_i can be mapped to $d(G_i) \subseteq U\mathbf{D}$. For robot x_i and given a robot x_j, we assume that

i) there is an edge from robot x_i to robot x_j;

ii) if there is a control influence on robot x_i from robot x_j, then we can denote such edge a control edge;

iii) the collection of all the control edges from all the robots in G_k^i is G_k^i's topology.

Under the limited sensing range, we can design control edges from robot x_i to its neighbours. However, it is unwise to simply assume that all the edges from robot x_i to all of its neighbours as robot x_i's control edges. Such a design would make maintaining these edges difficult and would induce additional computational load. Therefore, for G_k^i, we propose the minimal local topology below.

Definition 4.3.1. *(Minimal Local Topology) Assume that $\mathcal{E}$ is a group division operator. $\mathcal{E}^1A = A$ and there is a control edge between a_1, a_2 for $a_1, a_2 \in A$ if $n(A) = 2$. For any $a_1, a_2 \in A$, if a_1 and a_2 are connected with* *a control edge, then* $a_1 \underset{\mathcal{E}^\infty}{\sim} a_2$*; the topology* $\overrightarrow{A}$ *from* $\mathcal{E}^\infty A = \left\{ a_1 \quad a_2 \quad \ldots \right\}$ *is the set*

$$\overrightarrow{A} = \left\{ a_1^1 \underset{\mathcal{E}^\infty}{\sim} a_2^1 \quad a_1^2 \underset{\mathcal{E}^\infty}{\sim} a_2^2 \quad \ldots \right\}$$

where i) $\mathcal{E}^1 a' = a'$ *for any* $a' \in \mathcal{E}^\infty A$ *and ii)* $a'_1 \underset{\mathcal{E}^\infty}{\sim} a'_2 \in \overrightarrow{A}$ *for any* $a'_1, a'_2 \in a'$. *Then, the minimal local topology* $\overrightarrow{G_k^i}$ *from* $\mathcal{E}^\infty G_k^i$ *for* G_k^i *is defined as follows:*

The 0-th division:

$$\mathcal{E}^0 G_k^i = G_k^i.$$

The 1-st division:

$$\mathcal{E}^1 G_k^i = \left\{ \mathcal{E}_1^1 G_k^i \quad \mathcal{E}_2^1 G_k^i \right\}.$$

with $\mathcal{E}_1^1 G_k^i = \mathcal{E}^0 G_k^i \ominus \{x_{k_2}\}$ *and* $\mathcal{E}_2^1 G_k^i = \mathcal{E}^0 G_k^i \ominus \{x_{k_1}\}$*, where* $x_{k_1}, x_{k_2} \in \mathcal{E}^0 G_k^i$ *such that* $|x_{k_1} - x_{k_2}| \geq |x_{k'_1} - x_{k'_2}|$ *for any* $x_{k'_1}, x_{k'_2} \in \mathcal{E}^0 G_k^i$.

The 2-nd division:

$$\begin{aligned}
\mathcal{E}^2 G_k^i &= \mathcal{E}^1 \left\{ \mathcal{E}_1^1 G_k^i \quad \mathcal{E}_2^1 G_k^i \right\} \\
&= \left\{ \mathcal{E}^1 \mathcal{E}_1^1 G_k^i \quad \mathcal{E}^1 \mathcal{E}_2^1 G_k^i \right\} \\
&= \left\{ \mathcal{E}_1^1 \mathcal{E}_1^1 G_k^i \quad \mathcal{E}_2^1 \mathcal{E}_1^1 G_k^i \quad \mathcal{E}_1^1 \mathcal{E}_2^1 G_k^i \quad \mathcal{E}_2^1 \mathcal{E}_2^1 G_k^i \right\}.
\end{aligned}$$

with i) $\mathcal{E}_1^1 \mathcal{E}_1^1 G_k^i = \mathcal{E}_1^1 G_k^i \ominus \{x_{k_{12}}\}$ *and* $\mathcal{E}_2^1 \mathcal{E}_1^1 G_k^i = \mathcal{E}_1^1 G_k^i \ominus \{x_{k_{11}}\}$*, where* $x_{k_{11}}, x_{k_{12}} \in \mathcal{E}_1^1 G_k^i$ *such that* $|x_{k_{11}} - x_{k_{12}}| \geq |x_{k'_{11}} - x_{k'_{12}}|$ *for any* $x_{k'_{11}}, x_{k'_{12}} \in \mathcal{E}_1^1 G_k^i$*; ii)* $\mathcal{E}_1^1 \mathcal{E}_2^1 G_k^i = \mathcal{E}_2^1 G_k^i \ominus \{x_{k_{22}}\}$ *and* $\mathcal{E}_2^1 \mathcal{E}_2^1 G_k^i = \mathcal{E}_2^1 G_k^i \ominus \{x_{k_{21}}\}$*, where* $x_{k_{21}}, x_{k_{22}} \in \mathcal{E}_2^1 G_k^i$ *such that* $|x_{k_{21}} - x_{k_{22}}| \geq |x_{k'_{21}} - x_{k'_{22}}|$ *for any* $x_{k'_{21}}, x_{k'_{22}} \in \mathcal{E}_2^1 G_k^i$.

$\ldots \ldots$

The $(n(G_k^i)-1)$***-th division:***

$$\mathcal{E}^{n(G_k^i)-1}G_k^i = \Big\{\ \mathcal{E}_1^1\cdots\mathcal{E}_1^1\mathcal{E}_1^1G_k^i \quad \mathcal{E}_1^1\cdots\mathcal{E}_2^1\mathcal{E}_1^1G_k^i \quad \mathcal{E}_1^1\cdots\mathcal{E}_1^1\mathcal{E}_2^1G_k^i \quad \mathcal{E}_1^1\cdots\mathcal{E}_2^1\mathcal{E}_2^1G_k^i \\ \ldots \quad \mathcal{E}_2^1\cdots\mathcal{E}_2^1\mathcal{E}_2^1G_k^i\ \Big\}.$$

The ∞***-th division:***

$$\mathcal{E}^{\infty}G_k^i = \mathcal{E}^{n(G_k^i)-1}G_k^i.$$

with

$$\mathcal{E}^{\infty}G_k^i = \{\mathcal{E}_1^{\infty}G_k^i, \ldots, \mathcal{E}_{2^{n(G_k^i-2)}}^{\infty}G_k^i\}.$$

Remark 4.3.1. *Note that the control edges from a set G' include the control edge from x_i to x_j for any $x_i \neq x_j \in G'$. Therefore, if there are more than one elements in G', then the topology based on the edges is possibly redundant. As $\mathcal{E}^{\infty}G_k^i$ is defined based on relative displacements and there are only two elements in its members, the topology composed of the control edges from $\mathcal{E}^{\infty}G_k^i$ is the minimal local topology.*

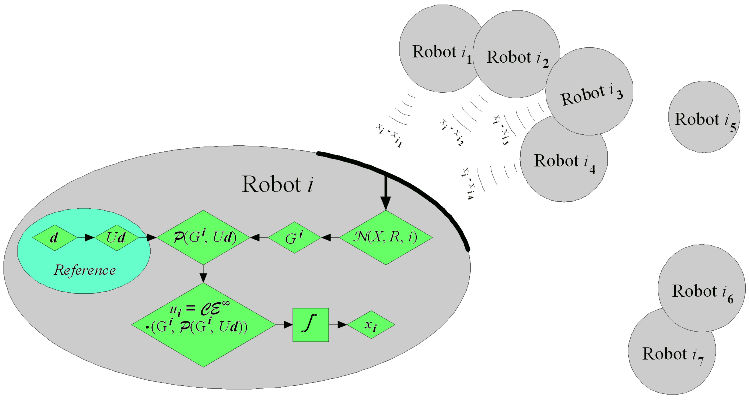


Figure 4.1: Demonstration sketch of robot i in interaction with other robots. Note that the distances in the figure are not to scale.

Since the edges that are the least possible to be disconnected are chosen as the control edges, the control law design based on $\mathcal{E}^{\infty}G_k^i$ is more robust. A fully connected G_k^i has $\frac{n(G_k^i)(n(G_k^i)-1)}{2}$ edges while depending on the structure of G_k^i, $\mathcal{E}^\infty G_k^i$ has far less control edges. Moreover, if we assume that the control topology is retained as long as the control edges in $\mathcal{E}^\infty G_k^i$ remain connected, then the computation burden could be effectively reduced.

Now, for robot x_i, we have defined 1) its neighbourhood as its information source; 2) groups of exclusive mutual neighbours where it can derive the available overlapped local agreements; 3) the mapping counterparts in the predefined formation; and 4) the minimal local topology. Then, under a control law $\mathcal{C}$ and a mapping protocol $\mathcal{P}$, we propose the first order system description below.

$$\begin{cases} \dot{x}_i = u_i, \\ u_i = \mathcal{C}\mathcal{E}^\infty(G^i, \mathcal{P}(G^i, U\mathbf{D})), \end{cases} \tag{4.7}$$

where i) $x_i \in \mathbf{R}^2$ and $u_i \in \mathbf{R}^2$ are the position and control force of robot x_i; ii) G^i is mapped to $\mathcal{P}(G^i, U\mathbf{D}) \in U\mathbf{D}$ by $\mathcal{P}$; iii) $\mathcal{E}^\infty(G^i, \mathcal{P}(G^i, U\mathbf{D}))$ includes the minimal local topologies from G^i and $U\mathbf{D}$; iv) $\mathcal{C}\mathcal{E}^\infty(G^i, \mathcal{P}(G^i, U\mathbf{D}))$ is the control force based on the minimal local topologies.

Under the control laws in (4.7), the interaction process of robot i is illustrated conceptually in Figure 4.1. Robots i_1, i_2, i_3, and i_4 are neighbours of robot i while robots i_5, i_6, and i_7 are not. Relative displacements of robot i's neighbours $x_i - x_{i_1}$, $x_i - x_{i_2}$, $x_i - x_{i_3}$, and $x_i - x_{i_4}$ are gathered from its measurement sensor to produce a collection G_i. Reference relative displacements are extracted from the predefined reference for group formation to produce $U\mathbf{D}$. Then, G_i is mapped to $U\mathbf{D}$ by the coordination protocol $d(G_i) = \mathcal{P}(G^i, U\mathbf{D})$ to generate the control force u_i. It can be seen that no data communication is needed in the process.

4.3.2 Control Law

Robots are autonomous, independent, and capable of making mapping decisions on their own. For robot x_i, local measurements are shared in a group of mutually exclusive neighbours G_k^i. Thus, such a group can serve as a reference for making mapping decisions. Since formation mappings are subject to changes and the mapping of a given instant is not really trustworthy, a bigger formation error does not deserve a bigger influence in the control force. We use the normalised sum to embody such considerations. Thus, we design the control law for robots in (4.8):

$$\begin{cases} \dot{x}_i = u_i, \\ u_i = \sum\limits_{G_k^i \in G_i} \sum\limits_{x_j \in G_k^i}^{x_j \underset{\mathcal{E}\infty}{\sim} x_i} \frac{-x_i + d^{i,k}(x_i) + x_j - d^{i,k}(x_j)}{|x_i - d^{i,k}(x_i) - x_j + d^{i,k}(x_j)|}, \end{cases} \tag{4.8}$$

where x_i is mapped to $d^{i,k}(x_i) \in \mathbf{D}$ in G_k^i by robot x_i; the group $\phi^{i,k}$ of control edges among robots in G_k^i is

$$(\phi^{i,k})^T = \left[\begin{array}{ccc} (\phi_1^{i,k})^T & (\phi_2^{i,k})^T & \dots \end{array} \right],$$

where $\phi_j^{i,k} = -x_i + d^{i,k}(x_i) + x_j - d^{i,k}(x_j)$ for any $x_j \in G_k^i$; G_k^i is mapped to $d(G_k^i) \subseteq \mathbf{D}$, where

$$d(G_k^i) = \left\{ \begin{array}{ccc} d^{i,k}(x_1^{i,k}) & d^{i,k}(x_2^{i,k}) & \dots \end{array} \right\}.$$

Thus, we can have

$$\begin{aligned} d(G_k^i) &= \ U_{k'}^{n(G_k^i)} \mathbf{D} \\ &= \{d^{i,k}(x_1^{i,k}), d^{i,k}(x_2^{i,k}), \dots, d^{i,k}(x_{n(G_k^i)}^{i,k})\}, \\ d(G_i) &= \{d(G_1^i), \dots, d(G_{n(G^i)}^i)\}, \\ M_{k,k'}^i &= [m_{ij}], \ \ m_{ij} = \begin{cases} 1, & \begin{array}{c} \text{if } d^{i,k}(x') = d' \\ \text{for } x' \in G_k^i, d' \in U_{k'}^{n(G_k^i)} \mathbf{D}; \end{array} \\ & \\ 0, & \quad \text{otherwise.} \end{cases} \end{aligned} \tag{4.9}$$

4.3.3 Event Triggered Coordination Protocol

We have given a general system description in (4.7) and proposed a control law in (4.8). In this section, we design the coordination protocol to provide the mapping from G_i to $\mathbf{D}$ in (4.9).

As discussed above in Section 4.3.1, the overlapped area is the key to a global consensus. Therefore, a mapping decision also needs to maximise the overlapped area. It can be seen that V^{G^i} in (4.10) can be used to evaluate the overlapped area around robot x_i:

$$V^{G^i} = \sum_{G^i_{k_1} \in G^i}^{k_1 \neq k_2} \sum_{G^i_{k_2} \in G^i} n(G^i_{k_1} \odot G^i_{k_2}). \tag{4.10}$$

Similarly, we can use (4.11) below to evaluate the overlapped area of the mapping decisions of robot x_i:

$$\begin{aligned} V^{d(G^i)} & \sum_{d(G^i_{k_1}) \in d(G^i)}^{k_1 \neq k_2} \sum_{d(G^i_{k_2}) \in d(G^i)} n(d(G^i_{k_1}) \odot d(G^i_{k_2})) \\ &= \sum_{d(G^i_{k_1}) \in d(G^i)}^{k_1 \neq k_2} \sum_{d(G^i_{k_2}) \in d(G^i)} \uplus(M^i_{k_1,k_1'} \circledast M^i_{k_2,k_2'}). \end{aligned} \tag{4.11}$$

where for $A = [a_{ij}], B = [b_{ij}] \in \mathbf{R}^{m \times n}$, $A \circledast B = [a_{ij} b_{ij}]$, and $\uplus(A) = \sum_i \sum_j a_{ij}$. On account of G^i's exclusive mutual neighbouring property, the eligible mapping decision should have a maximal $V^{d(G^i)}$. Moreover, the eligible mapping decisions should be compact and it is necessary that $d(G^i) \in \circledS$ in Definition 4.3.2. This consideration is especially helpful for robots on the edges to make mapping decisions.

Definition 4.3.2. *Assume that the compatible neighbor group for robot x_i is*

$$\boldsymbol{D}^i = \left\{ \begin{array}{ccc} d^i_1 & d^i_2 & \dots \end{array} \right\}$$

where $d^i_j \neq d^{i,k}(x_j)$ for any $d^i_j \in \boldsymbol{D}^i$, $G^i_k \in G^i$, and $x_j \in G^i_k$. If $|d^{i,k}(x_i) - d| > \max\{|d^{i,k}(x_i) - d'|, |d - d'|\}$ for any $d \in \boldsymbol{D} \ominus \boldsymbol{D}^i$, any $d' \in \boldsymbol{D}^i$, and any $G^i_k \in G^i$, then $d(G^i) \in \circledS$, which means that $d(G^i)$ is a collection of compact mapping decision sets in $\boldsymbol{D}$.

Then, we can find an optimal mapping decision that maximises the cross-correlation criterion in (4.12):

$$J^{G^i_k} = \sum_{x' \in G^i_k}^{x' \neq x''} \sum_{x'' \in G^i_k} \frac{(x' - x'')^T (d^{i,k}(x') - d^{i,k}(x''))}{|x' - x''| \cdot |d^{i,k}(x') - d^{i,k}(x'')|}. \tag{4.12}$$

Thus, robot x_i can make mapping decision $d^*(G^i)$ by (4.13) and (4.14). In (4.13), $\mathbf{D}^*(G_i)$ is a set of all the eligible mapping decisions. In (4.14), $d^*(G^i) \in \mathbf{D}^*(G_i)$ is the optimal mapping decision. This optimal mapping decision maps robot x_i to some point $d^*(x_i)$ in (4.15), and robot x_i's neighbours to point $d^*(x_i)$'s neighbours.

Protocol 4.3.1. *Robot x_i makes mapping decisions by (4.13), (4.14), and (4.15). If a different mapping decision is made, then we say that the protocol is validly triggered or activated; if otherwise, then it is not validly triggered or activated; if unspecified, then we only say that it is triggered or activated.*

$$\boldsymbol{D}^*(G_i) = \underset{d(G^i)\in \circledS}{\arg} \max_{d(G^i)\subseteq U\boldsymbol{D}} V^{d(G^i)}. \tag{4.13}$$

$$d^*(G^i) = \arg \max_{d(G^i)\in \boldsymbol{D}^*(G_i)} \sum_{G_k^i \in G^i} J^{G_k^i}. \tag{4.14}$$

$$d^*(x_i) = \arg \max_{d^*(x_i)\in \boldsymbol{D}} \sum_{G_k^i \in G^i}^{d^*(x_i)=d^{i,k}(x_i)} 1. \tag{4.15}$$

4.4 System Analysis

We have proposed our controller and protocol design. It is important that the robot team reaches some steady positions under the control law, and the collection of these steady positions can be justified to be a desired formation. In this section, we propose some analyses on these concerns.

4.4.1 Convergence Analysis

In this section, we show that the team of robots in (4.8) is driven to its steady state by the control law. Though the form of (4.8) is intuitive, it is not suitable for analysis. Thus, we rewrite robot x_i's control force as (4.16):

$$u_i = \sum_{G_k^i \in G_i} \theta^{i,k} |\phi^{i,k}|^{-1} \phi^{i,k} = \upsilon^{G^i} \Theta^{G^i} |\Phi^{G^i}|^{-1} \Phi^{G^i}, \tag{4.16}$$

where

$$
\begin{aligned}
\theta^{i,k} =& \mathbf{1}^T_{n(G^i_k)}, \;\; \upsilon^i = \mathbf{1}^T_{n(G^i)}, \\
\Theta^i =& \begin{bmatrix} \theta^{i,1} & & \emptyset \\ & \ddots & \\ \emptyset & & \theta^{i,kn(G^i)} \end{bmatrix}, \;\; \phi^{i,k} = \begin{bmatrix} \phi^{i,k}_1 \\ \vdots \\ \phi^{i,k}_{n(G^i_k)} \end{bmatrix}, \\
|\phi^{i,k}|^{-1} =& \begin{bmatrix} \frac{1}{|\phi^{i,k}_1|} & & 0 \\ & \ddots & \\ 0 & & \frac{1}{|\phi^{i,k}_{n(G^i_k)}|} \end{bmatrix}, \;\; \Phi^i = \begin{bmatrix} \phi^{i,1} \\ \vdots \\ \phi^{i,n(G^i)} \end{bmatrix}, \\
|\Phi^i|^{-1} =& \begin{bmatrix} |\phi^{i,1}|^{-1} & & \emptyset \\ & \ddots & \\ \emptyset & & |\phi^{i,n(G^i)}|^{-1} \end{bmatrix}.
\end{aligned}
$$

Then, we can rewrite (4.8) as (4.17):

$$
\begin{aligned}
\dot{X} =& \mathbf{u} = \Upsilon\Theta|\Phi|^{-1}\Phi, \\
\mathbf{u} =& \begin{bmatrix} u_1 \\ \vdots \\ u_N \end{bmatrix}, \;\; \Upsilon = \begin{bmatrix} \upsilon^1 & & \emptyset \\ & \ddots & \\ \emptyset & & \upsilon^N \end{bmatrix}, \\
\Theta =& \begin{bmatrix} \Theta^1 & & \emptyset \\ & \ddots & \\ \emptyset & & \Theta^N \end{bmatrix}, \;\; \Phi = \begin{bmatrix} \phi^1 \\ \vdots \\ \phi^N \end{bmatrix}, \\
|\Phi|^{-1} =& \begin{bmatrix} |\phi^1|^{-1} & & \emptyset \\ & \ddots & \\ \emptyset & & |\phi^N|^{-1} \end{bmatrix}.
\end{aligned}
\tag{4.17}
$$

There exists $L^{\mathbf{X}}$ and $L^{\mathbf{D}}$ such that $\Phi = L^{\mathbf{X}}X - L^{\mathbf{D}}D$. Then, by (4.17) and $\frac{d}{dt}(L^{\mathbf{D}}D) = 0$, we can have (4.18):

$$
\begin{aligned}
\dot{X} &= \Upsilon\Theta|\Phi|^{-1}L^{\mathbf{X}}(X - (L^{\mathbf{X}})^T(L^{\mathbf{X}}(L^{\mathbf{X}})^T)^{-1}L^{\mathbf{D}}D), \\
\dot{X}' &= \Upsilon\Theta|\Phi|^{-1}L^{\mathbf{X}}X', \\
X' &= X - (L^{\mathbf{X}})^T(L^{\mathbf{X}}(L^{\mathbf{X}})^T)^{-1}L^{\mathbf{D}}D,
\end{aligned}
\tag{4.18}
$$

$$|\Phi|^{-1}L^{\mathbf{X}} = |\Phi|^{-\frac{1}{2}}|\Phi|^{-\frac{1}{2}}L^{\mathbf{X}}|\Phi|^{\frac{1}{2}}|\Phi|^{-\frac{1}{2}}. \tag{4.19}$$

It can be seen in (4.19) that if $|\Phi|^{-\frac{1}{2}}L^{\mathbf{X}}|\Phi|^{\frac{1}{2}} \leq 0$, then $|\Phi|^{-1}L^{\mathbf{X}} \leq 0$. Meanwhile, since $|\Phi|^{-\frac{1}{2}}L^{\mathbf{X}}|\Phi|^{\frac{1}{2}}$ has the same eigenvalues with $L^{\mathbf{X}}$, and $L^{\mathbf{X}} \leq 0$, we have $|\Phi|^{-\frac{1}{2}}L^{\mathbf{X}}|\Phi|^{\frac{1}{2}} \leq 0$. Thus, it follows that

$$|\Phi|^{-1}L^{\mathbf{X}} = |\Phi|^{-\frac{1}{2}}|\Phi|^{-\frac{1}{2}}L^{\mathbf{X}}|\Phi|^{\frac{1}{2}}|\Phi|^{-\frac{1}{2}} \leq 0.$$

We then have $\Upsilon\Theta|\Phi|^{-1}L^{\mathbf{X}} \leq 0$. Thus, system (4.19) is stable and system (4.17) is stable as well.

We observe that (4.17) is stable, but we still have no clue of the properties of its steady states. Below, we show that at the steady states, at least $N-1$ elements of Φ are zero under a fixed mapping and a connected topology. By (4.17) and (4.18), we have

$$\begin{aligned}
\dot{\Phi} &= L^{\mathbf{X}}\dot{X} = L^{\mathbf{X}}\Upsilon\Theta|\Phi|^{-1}\Phi \\
&= L^{\mathbf{X}}\begin{bmatrix} \mathbf{1}^T_{\sum\limits_{1\leq i\leq n(G^1)} n(G_i^1)} & & \emptyset \\ & \ddots & \\ \emptyset & & \mathbf{1}^T_{\sum\limits_{1\leq i\leq n(G^N)} n(G_i^N)} \end{bmatrix}|\Phi|^{-1}\Phi, \\
\dot{\Phi} &= \mathbb{L}^{\mathbf{X}}|\Phi|^{-1}\Phi.
\end{aligned} \tag{4.20}$$

Since the topology is connected, then among the robots, there exist $N-1$ independent control edges that can linearly express any other control edges. Assume that these independent control edges correspond to $\hat{\Phi}$ where $n(\hat{\Phi}) = N-1$. Thus, there exist Γ_{Φ}, $\Lambda^{\hat{\Phi}}$ and $\Lambda^{\frac{D}{d(\hat{\Phi})}}$ in (4.21).

$$\begin{aligned}
\Phi &= \Gamma_{\Phi}\left(\begin{bmatrix} I \\ \Lambda^{\hat{\Phi}} \end{bmatrix}\hat{\Phi} + \begin{bmatrix} \emptyset \\ \Lambda^{\frac{D}{d(\hat{\Phi})}} \end{bmatrix} d(\hat{\Phi})\right) = \Gamma_{\Phi}\begin{bmatrix} I & \emptyset \\ \Lambda^{\hat{\Phi}} & \Lambda^{\frac{D}{d(\hat{\Phi})}} \end{bmatrix}\begin{bmatrix} \hat{\Phi} \\ d(\hat{\Phi}) \end{bmatrix}, \\
\Gamma_{\Phi} &= \Gamma_{\Phi}^T,\ \ \Gamma_{\Phi}^2 = I.
\end{aligned} \tag{4.21}$$

By (4.21), it follows that

$$|\Phi|^{-1}\Phi = \Gamma_{\Phi}\left(\begin{bmatrix} |\hat{\Phi}|^{-1} \\ |\Lambda^{\hat{\Phi}}\hat{\Phi}|^{-1}\Lambda^{\hat{\Phi}} \end{bmatrix}\hat{\Phi} + \begin{bmatrix} \emptyset \\ |\Lambda^{\hat{\Phi}}\hat{\Phi}|^{-1}\Lambda^{\frac{D}{d(\hat{\Phi})}} \end{bmatrix} d(\hat{\Phi})\right)$$

$$\begin{aligned}
&= \Gamma_\Phi \begin{bmatrix} |\hat{\Phi}|^{-1} & \emptyset \\ \emptyset & |\Lambda^{\hat{\Phi}}\hat{\Phi}|^{-1} \end{bmatrix} \left(\begin{bmatrix} I \\ \Lambda^{\hat{\Phi}} \end{bmatrix} \hat{\Phi} + \begin{bmatrix} \emptyset \\ \Lambda^{\frac{D}{d(\hat{\Phi})}} \end{bmatrix} d(\hat{\Phi}) \right) \\
&= \Gamma_\Phi \begin{bmatrix} |\hat{\Phi}|^{-1} & \emptyset \\ \emptyset & |\Lambda^{\hat{\Phi}}\hat{\Phi}|^{-1} \end{bmatrix} \begin{bmatrix} I & \emptyset \\ \Lambda^{\hat{\Phi}} & \Lambda^{\frac{D}{d(\hat{\Phi})}} \end{bmatrix} \begin{bmatrix} \hat{\Phi} \\ d(\hat{\Phi}) \end{bmatrix}, \\
\dot{\Phi} &= \Gamma_\Phi \begin{bmatrix} I \\ \Lambda^{\hat{\Phi}} \end{bmatrix} \dot{\hat{\Phi}} = \mathbb{L}^{\mathbf{X}} |\Phi|^{-1} \Phi. \qquad (4.22)
\end{aligned}$$

Then, by (4.20), (4.21) and (4.22), we have

$$\begin{aligned}
\dot{\hat{\Phi}} &= \begin{bmatrix} I & \emptyset \end{bmatrix} \Gamma_\Phi \mathbb{L}^{\mathbf{X}} |\Phi|^{-1} \Gamma_\Phi \left(\begin{bmatrix} I \\ \Lambda^{\hat{\Phi}} \end{bmatrix} \hat{\Phi} + \begin{bmatrix} \emptyset \\ \Lambda^{\frac{D}{d(\hat{\Phi})}} \end{bmatrix} d(\hat{\Phi}) \right) \\
&= \begin{bmatrix} I & \emptyset \end{bmatrix} \Gamma_\Phi \mathbb{L}^{\mathbf{X}} \Gamma_\Phi \begin{bmatrix} |\hat{\Phi}|^{-1} & \emptyset \\ \emptyset & |\Lambda^{\hat{\Phi}}\hat{\Phi}|^{-1} \end{bmatrix} \begin{bmatrix} I & \emptyset \\ \Lambda^{\hat{\Phi}} & \Lambda^{\frac{D}{d(\hat{\Phi})}} \end{bmatrix} \begin{bmatrix} \hat{\Phi} \\ d(\hat{\Phi}) \end{bmatrix}. \\
&\qquad (4.23)
\end{aligned}$$

We choose T in (4.24) as the energy function. By (4.20), (4.21), (4.22), and (4.23), we have

$$\begin{aligned}
T^G &= \frac{1}{2}(\Phi)^T \Phi, \qquad (4.24) \\
\dot{T}^G &= (\Phi)^T \dot{\Phi} \\
&= \left(\Gamma_\Phi \begin{bmatrix} I & \emptyset \\ \Lambda^{\hat{\Phi}} & \Lambda^{\frac{D}{d(\hat{\Phi})}} \end{bmatrix} \begin{bmatrix} \hat{\Phi} \\ d(\hat{\Phi}) \end{bmatrix} \right)^T \begin{bmatrix} I & \emptyset \end{bmatrix} \Gamma_\Phi \mathbb{L}^{\mathbf{X}} \Gamma_\Phi \\
&\quad \times \begin{bmatrix} |\hat{\Phi}|^{-1} & \emptyset \\ \emptyset & |\Lambda^{\hat{\Phi}}\hat{\Phi}|^{-1} \end{bmatrix} \begin{bmatrix} I & \emptyset \\ \Lambda^{\hat{\Phi}} & \Lambda^{\frac{D}{d(\hat{\Phi})}} \end{bmatrix} \begin{bmatrix} \hat{\Phi} \\ d(\hat{\Phi}) \end{bmatrix} \\
&= \left(\Gamma_\Phi \begin{bmatrix} I & \emptyset \\ \Lambda^{\hat{\Phi}} & \Lambda^{\frac{D}{d(\hat{\Phi})}} \end{bmatrix} \begin{bmatrix} |\hat{\Phi}| & \\ & I \end{bmatrix} \begin{bmatrix} |\hat{\Phi}|^{-1} & \\ & I \end{bmatrix} \begin{bmatrix} \hat{\Phi} \\ d(\hat{\Phi}) \end{bmatrix} \right)^T \begin{bmatrix} I & \emptyset \end{bmatrix} \\
&\quad \times \Gamma_\Phi \mathbb{L}^{\mathbf{X}} \Gamma_\Phi \begin{bmatrix} |\hat{\Phi}|^{-1} & \emptyset \\ \emptyset & |\Lambda^{\hat{\Phi}}\hat{\Phi}|^{-1} \end{bmatrix} \begin{bmatrix} I & \emptyset \\ \Lambda^{\hat{\Phi}} & \Lambda^{\frac{D}{d(\hat{\Phi})}} \end{bmatrix} \\
&\quad \times \begin{bmatrix} |\hat{\Phi}| & \\ & I \end{bmatrix} \begin{bmatrix} |\hat{\Phi}|^{-1} & \\ & I \end{bmatrix} \begin{bmatrix} \hat{\Phi} \\ d(\hat{\Phi}) \end{bmatrix} \\
&= \left(\begin{bmatrix} |\hat{\Phi}|^{-1} & \\ & I \end{bmatrix} \begin{bmatrix} \hat{\Phi} \\ d(\hat{\Phi}) \end{bmatrix} \right)^T \begin{bmatrix} |\hat{\Phi}| & \\ & I \end{bmatrix} \begin{bmatrix} I & (\Lambda^{\hat{\Phi}})^T \\ \emptyset & (\Lambda^{\frac{D}{d(\hat{\Phi})}})^T \end{bmatrix} \Gamma_\Phi \begin{bmatrix} I & \emptyset \end{bmatrix}
\end{aligned}$$

$$
\begin{aligned}
&\times\Gamma_{\Phi}\mathbb{L}^{\mathbf{X}}\Gamma_{\Phi}\begin{bmatrix} |\hat{\Phi}|^{-1} & \emptyset \\ \emptyset & |\Lambda^{\hat{\Phi}}\hat{\Phi}|^{-1} \end{bmatrix}\begin{bmatrix} |\hat{\Phi}| & \emptyset \\ \Lambda^{\hat{\Phi}}|\hat{\Phi}| & \Lambda^{\frac{D}{d(\hat{\Phi})}} \end{bmatrix} \\
&\times\left(\begin{bmatrix} |\hat{\Phi}|^{-1} & \\ & I \end{bmatrix}\begin{bmatrix} \hat{\Phi} \\ d(\hat{\Phi}) \end{bmatrix}\right) \\
={}&\left(\begin{bmatrix} |\hat{\Phi}|^{-1} & \\ & I \end{bmatrix}\begin{bmatrix} \hat{\Phi} \\ d(\hat{\Phi}) \end{bmatrix}\right)^{T}\begin{bmatrix} |\hat{\Phi}| & \emptyset \\ \Lambda^{\hat{\Phi}}|\hat{\Phi}| & \Lambda^{\frac{D}{d(\hat{\Phi})}} \end{bmatrix}^{T}\Gamma_{\Phi}\begin{bmatrix} I & \emptyset \end{bmatrix}\Gamma_{\Phi}\mathbb{L}^{\mathbf{X}}\Gamma_{\Phi} \\
&\times\begin{bmatrix} |\hat{\Phi}|^{-1} & \emptyset \\ \emptyset & |\Lambda^{\hat{\Phi}}\hat{\Phi}|^{-1} \end{bmatrix}\begin{bmatrix} |\hat{\Phi}| & \emptyset \\ \Lambda^{\hat{\Phi}}|\hat{\Phi}| & \Lambda^{\frac{D}{d(\hat{\Phi})}} \end{bmatrix}\left(\begin{bmatrix} |\hat{\Phi}|^{-1} & \\ & I \end{bmatrix}\begin{bmatrix} \hat{\Phi} \\ d(\hat{\Phi}) \end{bmatrix}\right) \\
={}&\left(\begin{bmatrix} |\hat{\Phi}|^{-1} & \\ & I \end{bmatrix}\begin{bmatrix} \hat{\Phi} \\ d(\hat{\Phi}) \end{bmatrix}\right)^{T}\left(\begin{bmatrix} |\hat{\Phi}|^{-1} & \emptyset \\ \emptyset & |\Lambda^{\hat{\Phi}}\hat{\Phi}|^{-1} \end{bmatrix}\times\begin{bmatrix} |\hat{\Phi}| & \emptyset \\ \Lambda^{\hat{\Phi}}|\hat{\Phi}| & \Lambda^{\frac{D}{d(\hat{\Phi})}} \end{bmatrix}\right)^{T} \\
&\begin{bmatrix} |\hat{\Phi}| & \emptyset \\ \emptyset & |\Lambda^{\hat{\Phi}}\hat{\Phi}| \end{bmatrix}\Gamma_{\Phi}\begin{bmatrix} I & \emptyset \end{bmatrix}\Gamma_{\Phi}\mathbb{L}^{\mathbf{X}}\Gamma_{\Phi}\left(\begin{bmatrix} |\hat{\Phi}|^{-1} & \emptyset \\ \emptyset & |\Lambda^{\hat{\Phi}}\hat{\Phi}|^{-1} \end{bmatrix}\right. \\
&\left.\times\begin{bmatrix} |\hat{\Phi}| & \emptyset \\ \Lambda^{\hat{\Phi}}|\hat{\Phi}| & \Lambda^{\frac{D}{d(\hat{\Phi})}} \end{bmatrix}\right)\left(\begin{bmatrix} |\hat{\Phi}|^{-1} & \\ & I \end{bmatrix}\begin{bmatrix} \hat{\Phi} \\ d(\hat{\Phi}) \end{bmatrix}\right) \\
={}&\left(\begin{bmatrix} |\hat{\Phi}|^{-1} & \\ & I \end{bmatrix}\begin{bmatrix} \hat{\Phi} \\ d(\hat{\Phi}) \end{bmatrix}\right)^{T}\begin{bmatrix} I & \emptyset \\ |\Lambda^{\hat{\Phi}}\hat{\Phi}|^{-1}\Lambda^{\hat{\Phi}}|\hat{\Phi}| & |\Lambda^{\hat{\Phi}}\hat{\Phi}|^{-1}\Lambda^{\frac{D}{d(\hat{\Phi})}} \end{bmatrix}^{T} \\
&\times\begin{bmatrix} |\hat{\Phi}| & \emptyset \\ \emptyset & |\Lambda^{\hat{\Phi}}\hat{\Phi}| \end{bmatrix}\Gamma_{\Phi}\begin{bmatrix} I & \emptyset \end{bmatrix}\Gamma_{\Phi}\mathbb{L}^{\mathbf{X}}\Gamma_{\Phi}\begin{bmatrix} I & \emptyset \\ |\Lambda^{\hat{\Phi}}\hat{\Phi}|^{-1}\Lambda^{\hat{\Phi}}|\hat{\Phi}| & |\Lambda^{\hat{\Phi}}\hat{\Phi}|^{-1}\Lambda^{\frac{D}{d(\hat{\Phi})}} \end{bmatrix} \\
&\times\left(\begin{bmatrix} |\hat{\Phi}|^{-1} & \\ & I \end{bmatrix}\begin{bmatrix} \hat{\Phi} \\ d(\hat{\Phi}) \end{bmatrix}\right)\leq 0. \qquad (4.25)
\end{aligned}
$$

As $\hat{\Phi}$ corresponds to groups of independent control edges, it follows that $\begin{bmatrix} I & \emptyset \\ |\Lambda^{\hat{\Phi}}\hat{\Phi}|^{-1}\Lambda^{\hat{\Phi}}|\hat{\Phi}| & |\Lambda^{\hat{\Phi}}\hat{\Phi}|^{-1}\Lambda^{\frac{D}{d(\hat{\Phi})}} \end{bmatrix}\times\left(\begin{bmatrix} |\hat{\Phi}|^{-1} & \\ & I \end{bmatrix}\begin{bmatrix} \hat{\Phi} \\ d(\hat{\Phi}) \end{bmatrix}\right)$ has non-zero independent elements. Thus, at the steady states where $\dot{T}=0$, by (4.25), we have

$$
\begin{bmatrix} |\hat{\Phi}| & \emptyset \\ \emptyset & |\Lambda^{\hat{\Phi}}\hat{\Phi}| \end{bmatrix}\Gamma_{\Phi}\begin{bmatrix} I & \emptyset \end{bmatrix}\Gamma_{\Phi}\mathbb{L}^{\mathbf{X}}\Gamma_{\Phi}=\emptyset.
$$

Then, it follows that

$$
\begin{bmatrix} |\hat{\Phi}| & \emptyset \\ \emptyset & |\Lambda^{\hat{\Phi}}\hat{\Phi}| \end{bmatrix}=\emptyset. \qquad (4.26)
$$

By (4.26), we have $\hat{\Phi} = \emptyset$. Thus, under a fixed mapping and a connected topology, system (4.8) is driven to a partially desired formation. Many researchers have focused on second-order systems [18, 26, 29, 32]. The model is first-order while it is applicable to second-order systems.

4.4.2 Existence and Uniqueness of Global Equilibrium

As shown in Section 4.4.1, system (4.8) can reach a steady state. However, the steady robot formation is not necessarily the predefined formation. In this section, we seek the condition that would make the steady state robot formation match the predefined formation.

First, we prove that under a justified predefined formation in the steady states, the norms of control edges are all zero. At the steady states, we have

$$\begin{aligned}
\Phi &= \hat{\Phi} + \hat{\Phi}' \\
&= \phi^G - H^G D, \\
\hat{\Phi} &= \phi_1^G - H_1^G D \\
&= \emptyset, \\
\hat{\Phi}' &= \phi_2^G - H_2^G D, \\
\phi^G &= \phi_1^G + \phi_2^G, \\
\phi_2^G &= F_1^G \phi_1^G, \\
\phi_1^G &= H_1^G D,
\end{aligned} \tag{4.27}$$

where $\mathbf{D}$ is known and needs to be justified. ϕ_1^G consists of independent elements. F_1^G and H_1^G are known and need to be tested to justify D. $\tilde{H}_1^G$ and H_2^G are unknown and ready to be solved and then used to test F_1^G and H_1^G. By (4.27), we have

$$\begin{aligned}
\phi_2^G &= F_1^G H_1^G D, \\
\phi^G &= (I + F_1^G) H_1^G D, \\
H^G D &= H_1^G D + H_{\mathbf{D}}^G \\
&= (H_1^G + H_2^G) D, \\
\Phi &= (I + F_1^G) H_1^G D - (H_1^G + H_2^G) D \\
&= (F_1^G H_1^G - H_2^G) D.
\end{aligned} \tag{4.28}$$

It follows that

$$\Phi = \begin{bmatrix} \Phi^{G^1_1} \\ \vdots \\ \Phi^{G^1_{n(G^1)}} \\ \Phi^{G^2_1} \\ \vdots \\ \Phi^{G^2_{n(G^2)}} \\ \vdots \\ \Phi^{G^N_1} \\ \vdots \\ \Phi^{G^N_{n(G^N)}} \end{bmatrix}, \; \Gamma_\Phi = \begin{bmatrix} \Gamma_{\Phi^{G^1_1}} \\ \vdots \\ \Gamma_{\Phi^{G^1_{n(G^1)}}} \\ \Gamma_{\Phi^{G^2_1}} \\ \vdots \\ \Gamma_{\Phi^{G^2_{n(G^2)}}} \\ \vdots \\ \Gamma_{\Phi^{G^N_1}} \\ \vdots \\ \Gamma_{\Phi^{G^N_{n(G^N)}}} \end{bmatrix}.$$

Then, we have

$$\begin{aligned} \Phi^{G^i} &= (F_1^{G^i} H_1^{G^i} - H_2^{G^i})D, \\ \phi^{G^i} &= (I + F_1^{G^i})H_1^{G^i} D, \\ \phi_e^{G^i} &= E_{\mathbf{X}}^{G^i} \phi^{G^i} = E_{\mathbf{X}}^{G^i}(I + F_1^{G^i})H_1^{G^i} D = \hat{E}_{\mathbf{X}}^{G^i} D, \\ \hat{E}_{\mathbf{X}}^{G^i} &= E_{\mathbf{X}}^{G^i}(I + F_1^{G^i})H_1^{G^i}, \end{aligned} \tag{4.29}$$

where $E_{\mathbf{X}}^{G^i}$ is used to extend ϕ^{G^i} and $\phi_e^{G^i}$ is the extended version of ϕ^{G^i}, as Φ is from minimal local topologies according to (4.8) and Definition 4.3.1. The same is in (4.30). Also, $E_{\mathbf{X}}^{G^i}$ is dependent on ϕ^G. Since $\phi^G = (I + F_1^G)H_1^G D$ and $\mathbf{D}$ has fixed known values, then $E_{\mathbf{X}}^{G^i}$ is dependent on F_1^G and H_1^G. Similarly, we have

$$\begin{aligned} H^{G^i} D &= (H_1^{G^i} + H_2^{G^i})D, \\ H_e^{G^i} D &= E_{\mathbf{D}}^{G^i} H^{G^i} D = E_{\mathbf{D}}^{G^i}(H_1^{G^i} + H_2^{G^i})D, \\ H_e^{G^i} &= E_{\mathbf{D}}^{G^i}(H_1^{G^i} + H_2^{G^i}), \end{aligned} \tag{4.30}$$

where $E_{\mathbf{D}}^{G^i}$ is dependent on $\mathbf{D}$. Since $\mathbf{D}$ has fixed known values, then $E_{\mathbf{D}}^{G^i}$ has fixed known values. Thus, by (4.12), (4.29) and (4.30), we have

$$\begin{aligned} \tilde{J}^{G^i} &= D^T(\tilde{E}_{\mathbf{X}}^{G^i})^T \left(\tilde{E}_{\mathbf{X}}^{G^i} D\right)^{-1} \left(\tilde{H}_e^{G^i} D\right)^{-1} \tilde{H}_e^{G^i} D, \\ \tilde{E}_{\mathbf{X}}^{G^i} &= E_{\mathbf{X}}^{G^i}(I + F_1^{G^i})\tilde{H}_1^{G^i}, \\ \tilde{H}_e^{G^i} &= E_{\mathbf{D}}^{G^i}(\tilde{H}_1^{G^i} + H_2^{G^i}). \end{aligned} \tag{4.31}$$

Assume that by (4.31), it follows that

$$\begin{aligned}\{\tilde{H}_1^{*G_i}, H_2^{*G_i}\} &= \arg\min_{\tilde{H}_1^G, H_2^G} \tilde{J}^{G^i}, \\ \tilde{H}_1^{*G} &= \tilde{H}_1^{*G_1} + \tilde{H}_1^{*G_2} + \cdots + \tilde{H}_1^{*G_N}. \qquad (4.32)\end{aligned}$$

Now we can discuss how the predefined formation is justified by (4.32). For a team of robots at steady states, Statements 4.4.1 and 4.4.2 can be used to describe the system.

Statement 4.4.1. *If the team of robots has reached their steady states and the mappings only trigger Protocol 4.3.1 in an invalid way, then the steady state formation is the predefined formation. In other words, in the process where certain $n-1$ control edges converge to zero, formation mismatches triggers Protocol 4.3.1 in a valid way.*

Statement 4.4.2. *If the team of robots has reached their steady states and the mappings only trigger Protocol 4.3.1 in an invalid way, then the steady state formation may not be the predefined formation. In other words, in the process where certain $n-1$ control edges converge to zero, formation mismatches do not validly trigger Protocol 4.3.1.*

Note that if $\tilde{H}_1^{*G} \neq H_1^G$, then $\mathbf{D}$ is justified and the robot formation is not steady and validly triggers Protocol 4.3.1; if $\tilde{H}_1^{*G} = H_1^G$ and $F_1^G H_1^{*G} - H_2^{*G} = \emptyset$, then $\mathbf{D}$ is justified; if otherwise, then $\mathbf{D}$ is not justified. For all admissible matrices of F_1^G and H_1^G, if $\mathbf{D}$ is always justified, then under $\mathbf{D}$, Statement 4.4.1 is true; if otherwise, then under $\mathbf{D}$, Statement 4.4.1 is true. Assume that under $\mathbf{D}$, Statement 4.4.1 is true. At the steady states, we have $\tilde{H}_1^{*G} = H_1^G$ and $F_1^G \tilde{H}_1^{*G} - H_2^{*G} = \emptyset$. It follows that $\Phi = (F_1^G H_1^{*G} - H_2^{*G})D = \emptyset$.

Below, we prove that if, within the sensing range R, the team of robots reaches a steady state where robots are no longer subject to changes in their mappings by triggering Protocol 4.3.1, then the same is with a team of robots under sensing range $R' > R$.

Small enlargement of the sensing range triggers only a small change. There are overlapped parts among elements in G^i and G and under different sensing ranges, these overlapped parts differ. To derive the relationships among steady mappings under different sensing ranges, Assumption 4.4.1 is made.

Assumption 4.4.1. *Under sensing range R, for any $G^{i_1}_{k^R_1}, G^{i_2}_{k^R_2} \in G_R$, if $i_1 \neq i_2$, then $G^{i_1}_{k^R_1} \neq G^{i_2}_{k^R_2}$; if $i_1 = i_2$ and $k^R_1 \neq k^R_2$, then $G^{i_1}_{k^R_1} \neq G^{i_2}_{k^R_1}$.*

Note that G_R, $G^i_{k^R_{j_1}}$ and $G^i_{k^R_{j_2}}$ are defined under R; $G_{R'}$ and $G^i_{k^{R'}}$ are defined under R'. Assume that for robot x_i, $x_{j_1} \in G^i_{k^R_{j_1}}$, $x_{j_1} \notin G^i_{k^R_{j_2}}$, $x_{j_2} \in G^i_{k^R_{j_2}}$, $x_{j_2} \notin G^i_{k^R_{j_1}}$, and $G^i_{k^R_{j_1}} \ominus \{x_{j_1}\} = G^i_{k^R_{j_2}} \ominus \{x_{j_2}\}$ under R; $G^i_{k^{R'}} = G^i_{k^R_{j_1}} \oplus \{x_{j_2}\}$ under R'. Moreover, $G_R \ominus G^i_{k^R_{j_1}} \ominus G^i_{k^R_{j_2}} = G_{R'} \ominus G^i_{k^{R'}}$.

Under sensing range R, for any robot x_j in $G^i_{k^R_{j_1}} \ominus \{x_{j_1}\}$, it is able to access both $G^i_{k^R_{j_1}}$ and $G^i_{k^R_{j_2}}$ and map them to adjacent points in $\mathbf{D}$. From robot x_j's mapping decision $d(G^i_{k^R_{j_1}})$ and $d(G^i_{k^R_{j_2}})$, we have maximised $J^{G^i_{k^R_{j_1}}}$ and $J^{G^i_{k^R_{j_2}}}$. Since $d(G^i_{k^R_{j_1}})$ and $d(G^i_{k^R_{j_2}})$ are unique and adjacent, these two mappings are also admissible optimal mapping decisions for x_{j_1} and x_{j_2}.

Under sensing range R', for any robot x_j in $G^i_{k^R_{j_1}} \ominus \{x_{j_1}\}$, it is able to access $G^i_{k^{R'}}$. As $G^i_{k^R_{j_1}} \oplus G^i_{k^R_{j_2}} = G^i_{k^{R'}}$ and $d(G^i_{k^R_{j_1}})$ and $d(G^i_{k^R_{j_2}})$ are unique and adjacent, robot x_j can make the same decision with that under sensing range R. Since $G_R \ominus G^i_{k^R_{j_1}} \ominus G^i_{k^R_{j_2}} = G_{R'} \ominus G^i_{k^{R'}}$, the rest of the robot formation and its mappings stay the same. To sum up, the mapping under sensing range R is the same with that under R'.

Assuming that the mappings are the same for R and R' and the robots are using minimal control topology. We have that if for robot x_i, $u_i = 0$ under R, then $u_i = 0$ under R'. It follows that if the robot formation is steady under R, then it is steady under R'.

We can do the same deduction under larger sensing range than R'. It can be seen that the mapping decision under R is the same with that under sufficiently large sensing ranges where $G_{\infty^+} = \mathbf{X}$. Since the mapping of $\mathbf{X}$ with $G_{\infty^+} = \mathbf{X}$ is a one-to-one mapping with $\mathbf{D}$, the predefined formation $\mathbf{D}$ is completely reached.

To sum up, in (4.32), if $\tilde{H}^{*G}_1 \neq H^G_1$, then $\mathbf{D}$ is justified and the robot formation is not steady and validly triggers Protocol 4.3.1; if $\tilde{H}^{*G}_1 = H^G_1$ and $F^G_1 H^{*G}_1 - H^{*G}_2 = \emptyset$, then $\mathbf{D}$ is justified; if otherwise, then $\mathbf{D}$ is not justified.

4.4.3 Consensus at the Predefined Formation

In Section 4.4.2, we proposed a method to justify the predefined formation **D**. Once **D** is justified, we need to show that Protocol 4.3.1 is able to transform the robot formation to **D**. Assume that there are L_{ϕ^G}, L_{H^G} and L_{ϕ^*}, and independent ϕ^* and H^*D so that we can have

$$\Phi = L_{\phi^G}\phi^* - L_{H^G}H^*D, \;\; \Delta\dot{X}^* = L_{\phi^*}\dot{X}. \tag{4.33}$$

By (4.27) and (4.33), it follows that

$$\begin{aligned}\phi^G =& L_{\phi^G}\phi^*, \;\; H^GD = L_{H^G}H^*D \\ \Delta\dot{X}^* =& L_{\phi^*}\dot{X} \\ =& L_{\phi^*}\Upsilon\Theta|\Phi|^{-1}L_{\phi^G}(\phi^* - H^*D) + L_{\phi^*}\Upsilon\Theta|\Phi|^{-1}(L_{\phi^G} - L_{H^G})H^*D\end{aligned} \tag{4.34}$$

Assume that $\epsilon = \phi^* - H^*D$ and $\dot{\epsilon} = \Delta\dot{X}^*$. By (4.34), we have

$$\begin{aligned}\dot{\epsilon} =& L_{\phi^*}\Upsilon\Theta|\Phi|^{-1}L_{\phi^G}\epsilon + L_{\phi^*}\Upsilon\Theta \\ & \times|\Phi|^{-1}(L_{\phi^G} - L_{H^G})H^*D.\end{aligned} \tag{4.35}$$

It can be seen from (4.35) that if $|(L_{\phi^G} - L_{H^G})H^*D|$ converges to 0 as ϵ converges, then the robot team reaches the predefined formation. Similarly with (4.29) and (4.30), we have

$$\begin{aligned}J^{G^i} &= \phi^{*T}L^T_{\phi^{G^i}}(E^{G^i}_{\mathbf{X}})^T|\phi^e_{G^i}|^{-1}|H^e_{G^i}D|^{-1}E^{G^i}_{\mathbf{D}}L_{H^{G^i}}H^*D \\ \phi^e_{G^i} &= E^{G^i}_{\mathbf{X}}L_{\phi^{G^i}}\phi^*, \\ H^e_{G^i} &= E^{G^i}_{\mathbf{D}}L_{H^{G^i}}H^*.\end{aligned} \tag{4.36}$$

By (4.36), it follows that

$$J^G = \phi^{*T}L^T_{\phi^G}(E^G_{\mathbf{X}})^T|\phi^e_G|^{-1}|H^e_GD|^{-1}E^G_{\mathbf{D}}L_{H^G}H^*D \tag{4.37}$$

Assume that Protocol 4.3.1 is activated at instant t, and t^- and t^+ are instants immediately before and after instant t. Thus, $J^G = J^G(t^-)$ is before the activation of Protocol 4.3.1, $\hat{J}^G = J^G(t^+)$ is after, and $J^G < \hat{J}^G$. We have

$$\begin{aligned}
&\phi^{*T} L_{\phi^G}^T (E_{\mathbf{X}}^G)^T |\phi_G^e|^{-1} |H_G^e D|^{-1} E_{\mathbf{D}}^G L_{H^G} H^* D \\
&< \phi^{*T} L_{\phi^G}^T (E_{\mathbf{X}}^G)^T |\phi_G^e|^{-1} |\hat{H}_G^e D|^{-1} \hat{E}_{\mathbf{D}}^G \hat{L}_{H^G} \hat{H}^* D, \\
&\phi^{*T} L_{\phi^G}^T (E_{\mathbf{X}}^G)^T |\phi_G^e|^{-1} \Big(|\hat{H}_G^e D|^{-1} \hat{E}_{\mathbf{D}}^G \hat{L}_{H^G} \hat{H}^* D \\
&- |H_G^e D|^{-1} E_{\mathbf{D}}^G L_{H^G} H^* D \Big) > 0. \qquad (4.38)
\end{aligned}$$

Rewrite (4.38), and we have

$$\begin{aligned}
&\mathbb{L}_{\phi^G}^T \quad (\hat{\mathbb{L}}_{H^G} - \mathbb{L}_{H^G}) > 0, \\
&\mathbb{L}_{\phi^G} = |\phi_G^e|^{-1} E_{\mathbf{X}}^G L_{\phi^G} L_{\phi^*} X, \\
&\hat{\mathbb{L}}_{H^G} = |\hat{H}_G^e D|^{-1} \hat{E}_{\mathbf{D}}^G \hat{L}_{H^G} \hat{H}^* D. \qquad (4.39)
\end{aligned}$$

It follows that

$$\begin{aligned}
&(\mathbb{L}_{\phi^G} - \hat{\mathbb{L}}_{H^G})^2 - (\mathbb{L}_{\phi^G} - \mathbb{L}_{H^G})^2 \\
&= \mathbb{L}_{\phi^G}^2 + \hat{\mathbb{L}}_{H^G}^2 - 2\mathbb{L}_{\phi^G}^T \hat{\mathbb{L}}_{H^G} - \mathbb{L}_{\phi^G}^2 - \mathbb{L}_{H^G}^2 + 2\mathbb{L}_{\phi^G}^T \mathbb{L}_{H^G} \\
&= n(\mathbb{L}_{\phi^G}) + n(\hat{\mathbb{L}}_{H^G}) - 2\mathbb{L}_{\phi^G}^T \hat{\mathbb{L}}_{H^G} - n(\mathbb{L}_{\phi^G}) - n(\mathbb{L}_{H^G}) + 2\mathbb{L}_{\phi^G}^T \mathbb{L}_{H^G} (4.40)
\end{aligned}$$

Since both $\hat{\mathbb{L}}_{H^G}$ and $\mathbb{L}_{H^G}$ have the same size with $\mathbb{L}_{\phi^G}$, we have $n(\mathbb{L}_{\phi^G}) = n(\hat{\mathbb{L}}_{H^G})$ and $n(\mathbb{L}_{\phi^G}) = n(\mathbb{L}_{H^G})$. By (4.40), we have

$$\begin{aligned}
&(\mathbb{L}_{\phi^G} - \hat{\mathbb{L}}_{H^G})^2 - (\mathbb{L}_{\phi^G} - \mathbb{L}_{H^G})^2 \\
&= -2\mathbb{L}_{\phi^G}^T (\hat{\mathbb{L}}_{H^G} - \mathbb{L}_{H^G}) < 0, \\
&(\mathbb{L}_{\phi^G} - \hat{\mathbb{L}}_{H^G})^2 < (\mathbb{L}_{\phi^G} - \mathbb{L}_{H^G})^2. \qquad (4.41)
\end{aligned}$$

By (4.41), we can see that $|\mathbb{L}_{\phi^G} - \mathbb{L}_{H^G}|$ is decreasing at the activation of Protocol 4.3.1. Then, it follows that

$$\begin{aligned}
&\mathbb{L}_{\phi^G} - \mathbb{L}_{H^G} \\
&= |\phi_G^e|^{-1} E_{\mathbf{X}}^G L_{\phi^G} L_{\phi^*} X - |H_G^e D|^{-1} E_{\mathbf{D}}^G L_{H^G} H^* D \\
&= |\phi_G^e|^{-1} E_{\mathbf{X}}^G L_{\phi^G} (L_{\phi^*} X - H^* D + H^* D) - |H_G^e D|^{-1} E_{\mathbf{D}}^G L_{H^G} H^* D \\
&= |\phi_G^e|^{-1} E_{\mathbf{X}}^G L_{\phi^G} (L_{\phi^*} X - H^* D) + (|\phi_G^e|^{-1} E_{\mathbf{X}}^G L_{\phi^G} - |H_G^e D|^{-1} E_{\mathbf{D}}^G L_{H^G}) H^* D \\
&= |\phi_G^e|^{-1} E_{\mathbf{X}}^G L_{\phi^G} (L_{\phi^*} X - H^* D) - |\phi_G^e|^{-1} (E_{\mathbf{D}}^G - E_{\mathbf{X}}^G) L_{H^G} H^* D
\end{aligned}$$

$$
\begin{aligned}
&-(|H_G^e D|^{-1} - |\phi_G^e|^{-1})(E_{\mathbf{D}}^G - E_{\mathbf{X}}^G)L_{H^G}H^*D \\
&+|\phi_G^e|^{-1}E_{\mathbf{X}}^G(L_{\phi^G} - L_{H^G})H^*D \\
=\ & \epsilon_1 + \epsilon_2 + \epsilon_3 + \epsilon_4, \qquad (4.42)
\end{aligned}
$$

where

$$
\begin{aligned}
\epsilon_1 &= |\phi_G^e|^{-1}E_{\mathbf{X}}^G L_{\phi^G}(L_{\phi^*}X - H^*D), \\
\epsilon_2 &= -|\phi_G^e|^{-1}(E_{\mathbf{D}}^G - E_{\mathbf{X}}^G)L_{H^G}H^*D, \\
\epsilon_3 &= -(|H_G^e D|^{-1} - |\phi_G^e|^{-1})(E_{\mathbf{D}}^G - E_{\mathbf{X}}^G)L_{H^G}H^*D, \\
\epsilon_4 &= |\phi_G^e|^{-1}E_{\mathbf{X}}^G(L_{\phi^G} - L_{H^G})H^*D.
\end{aligned}
$$

Note that $|\phi_G^e|^{-1}$, $E_{\mathbf{X}}^G$, L_{ϕ^G}, and $L_{\phi^*}X$ stay the same before and after each activation of Protocol 4.3.1. $E_{\mathbf{X}}^G$ and $|\phi_G^e|^{-1}$ are decided by L_{ϕ^G}, and $E_{\mathbf{D}}^G$ and $|H_G^e D|^{-1}$ are decided by $L_{H^G}H^*D$. As $|L_{\phi^*}X - H^*D|$ converges, ϵ_1 also converges. Once Protocol 4.3.1 is validly activated to make a new mapping decision and a smaller $|\mathbb{L}_{\phi^G} - \mathbb{L}_{H^G}|$ is generated, a smaller $|\epsilon_2 + \epsilon_3 + \epsilon_4|$ is also favoured.

It can be seen in (4.42) that $\|L_{\phi^G} - L_{H^G}\|$ is bounded by $|\epsilon_2|$, $|\epsilon_3|$ and $|\epsilon_4|$. As $|\epsilon_2|$, $|\epsilon_3|$ and $|\epsilon_4|$ converge and $|\mathbb{L}_{\phi^G} - \hat{\mathbb{L}}_{H^G}|$ decreases to zero, $\|L_{\phi^G} - L_{H^G}\|$ gradually converges to zero as well.

4.4.4 Probability Based Protocol

If Protocol 4.3.1 was constantly active, it would cause some heavy computation burden at large system scales. Most of the time, however, it would not be validly triggered. For one thing, active Protocol 4.3.1 does not necessarily induce a new mapping decision. For another, it has been shown in Sections 4.4.2 and 4.4.3 that Protocol 4.3.1 is already capable of driving the steady states of a robot team to the predefined formation. Thus, robots can identify local steady states and deactivate Protocol 4.3.1 to reduce computation burden. However, since the multi-robot system is distributed and localised and there is no global monitor or clock, it is impossible to regulate the robots in such a manner. Therefore, to deal with the problem, we use probability theory to derive a new coordination protocol, Protocol 4.4.1.

Protocol 4.4.1. *The possibility that at the current instant, robot x_i makes mapping decisions by (4.13), (4.14) and (4.15) is p, where* $0 < p < 1$.

Note that for Protocol 4.4.1, p_i is the possibility that it is not activated at instant k by robot x_i. Then, for robot x_i, $t^i_{av} = \tau \lim_{k\to\infty} \frac{k(1-p_i)}{kp_i}$ is the average time difference between two activations of Protocol 4.4.1, where τ is the sampling time between two consecutive instants. Thus, in order for the activation of Protocol 4.4.1 to be only at the steady states, t^i_{av} needs to be longer than the wandering time for robot x_i to reach its local steady states with allowable deviation. Moreover, the longer t^i_{av} is, the more reduced the computation is and the slower the robot team reaches a global consensus. Thus, a shorter t^i_{av} is optional to achieve a trade-off. However, it is observed in the simulation study that if t^i_{av} is too short and Protocol 4.4.1 is validly triggered sufficiently frequently, robot x_i may be trapped in an infinite loop induced by Protocol 4.4.1. Further research can derive the radius of such protocol-induced loops and find a lower bound on t^i_{av}.

4.4.5 Connectivity Maintenance

The convergence analysis in Section 4.4.1 assumed that system (4.8) is under a connected topology. In this section, we propose a condition that ensures system connectivity.

Assume that $t = t_k$ is the instant when the mapping of the robot team is changed at the k-th time. Thus, $t = t_1$ is the initial (zero-th) instant, before which the robots have no mapping decisions, and after which the robots start to move; the robot team has a fixed mapping at $t_k \le t < t_{k+1}$. Thus, G_{t_1} is the control edge at $t_1 \le t < t_2$, and G_{t_k} is the control edge at $t_k \le t < t_{k+1}$. At $t = t_k^-$, $k > 2$, the robot team has converged to a steady state under control edge $G^{t_{k-1}}$. Since the system has a finite size and the steady state properties have been derived in Section 4.4.1, there exists a known set $\mathcal{G}_{\tau^-}$ to include all admissible steady state control edges under a given $\mathbf{D}$.

Assume that of all the connected robots, x_i and x_j are furtherest apart. According to (4.24), we have

$$T^G = \frac{1}{2}(\Phi)^T\Phi.$$

It is shown that under a fixed and connected topology, T^G is non-increasing. At $t_1 \le t < t_2$

$$\begin{aligned} T^{G_{t_1}} &\ge (x_i - x_j - d^*(x_i) + d^*(x_j))^2, \\ |x_i - x_j| &\le \sqrt{T^{G_{t_1}}} + |d^*(x_i) - d^*(x_j)| \end{aligned}$$

$$\leq \sqrt{T^{G_{t_1}}} + \max\{|\mathcal{E}^{\infty}\mathbf{D}|\}. \tag{4.43}$$

At $t \geq t_2$, we have

$$\begin{aligned}
T^{G_t} &\geq (x_i - x_j - d^*(x_i) + d^*(x_j))^2, \\
\sqrt{T^{G_t}} &\in \sqrt{T^{\mathcal{G}_{\tau^-}}}, \\
|x_i - x_j| &\leq \sqrt{T^{G_t}} + |d^*(x_i) - d^*(x_j)| \\
&\leq \max\{\sqrt{T^{\mathcal{G}_{\tau^-}}}\} + \max\{|\mathcal{E}^{\infty}\mathbf{D}|\}.
\end{aligned} \tag{4.44}$$

Thus, by (4.43) and (4.44), at a sufficiently long t^i_{av}, if (4.45) is satisfied and all the robots behave according to Protocol 4.4.1, then system (4.8) always stays connected.

$$\begin{aligned}
R &\geq \max\{\sqrt{T^{G_{t_1}}}, \sqrt{T^{\mathcal{G}_{\tau^-}}}\} + \max\{|\mathcal{E}^{\infty}\mathbf{D}|\}, \\
\mathcal{G}_{\tau^-} &= \{\mathcal{G}_{\tau_1^-}, \mathcal{G}_{\tau_2^-}, \ldots, \mathcal{G}_{\tau^-_{n(\mathcal{G}_{\tau^-})}}\}, \\
\sqrt{T^{\mathcal{G}_{\tau^-}}} &= \{\sqrt{T^{\mathcal{G}_{\tau_1^-}}}, \sqrt{T^{\mathcal{G}_{\tau_2^-}}}, \ldots, \sqrt{T^{\mathcal{G}_{\tau^-_{n(\mathcal{G}_{\tau^-})}}}}\}, \\
\mathcal{E}^{\infty}\mathbf{D} &= \{\mathcal{E}^{\infty}_1\mathbf{D}, \mathcal{E}^{\infty}_2\mathbf{D}, \ldots, \mathcal{E}^{\infty}_{2^{n(\mathbf{D})-2}}\mathbf{D}\} \\
\mathcal{E}^{\infty}_l\mathbf{D} &= \left\{ \begin{array}{ccc} d_1^{\mathcal{E}^{\infty}_l\mathbf{D}} & d_2^{\mathcal{E}^{\infty}_l\mathbf{D}} & \ldots \end{array} \right\}, \\
|\mathcal{E}^{\infty}\mathbf{D}| &= \left\{ \begin{array}{ccc} |\mathcal{E}^{\infty}_1\mathbf{D}| & |\mathcal{E}^{\infty}_2\mathbf{D}| & \ldots \end{array} \right\}, \\
|\mathcal{E}^{\infty}_l\mathbf{D}| &= \left\{ \begin{array}{cccccc} |\delta d_{12}^{\mathcal{E}^{\infty}_l\mathbf{D}}| & |\delta d_{13}^{\mathcal{E}^{\infty}_l\mathbf{D}}| & \ldots & |\delta d_{23}^{\mathcal{E}^{\infty}_l\mathbf{D}}| & |\delta d_{24}^{\mathcal{E}^{\infty}_l\mathbf{D}}| & \ldots \end{array} \right\},
\end{aligned} \tag{4.45}$$

where $\delta d_{l_1 l_2}^{\mathcal{E}^{\infty}_l\mathbf{D}} = d_{l_1}^{\mathcal{E}^{\infty}_l\mathbf{D}} - d_{l_2}^{\mathcal{E}^{\infty}_l\mathbf{D}}$ for any $d_{l_1}^{\mathcal{E}^{\infty}_l\mathbf{D}}, d_{l_2}^{\mathcal{E}^{\infty}_l\mathbf{D}} \in \mathcal{E}^{\infty}_l\mathbf{D} \in \mathcal{E}^{\infty}\mathbf{D}$.

It can be seen in (4.45) that due to the reduced number of control edges by Definition 4.3.1, T^G is reduced effectively. Also, since a minimal topology is used, only a small bias of $\max\{|\mathcal{E}^{\infty}\mathbf{D}|\}$ is enough to maintain system connectivity.

4.5 Simulation Examples

In this section, simulations are given to demonstrate the effectiveness of the proposed strategy. The sensing range is $R = 35$ meters. Two teams of eight and five robots, $\mathbf{X}_1(t_1)$ and $\mathbf{X}_2(t_1)$, and their corresponding predefined formations, $\mathbf{D}_1$ and $\mathbf{D}_2$, are initiated in (4.46) and (4.47).

$$\begin{aligned}
\mathbf{X}_1(t_1) &= \left\{ \begin{bmatrix} -0.697 \\ -36.452 \end{bmatrix}, \begin{bmatrix} -30.155 \\ -30.155 \end{bmatrix}, \begin{bmatrix} -2.152 \\ -42.812 \end{bmatrix}, \begin{bmatrix} 36.501 \\ -43.324 \end{bmatrix}, \right. \\
&\qquad \left. \begin{bmatrix} 26.866 \\ -40.747 \end{bmatrix}, \begin{bmatrix} 15.749 \\ -52.5 \end{bmatrix}, \begin{bmatrix} -2.907 \\ -13.579 \end{bmatrix}, \begin{bmatrix} 4.732 \\ 4.196 \end{bmatrix} \right\}, \\
\mathbf{D}_1 &= \left\{ \begin{bmatrix} 31.917 \\ 17.622 \end{bmatrix}, \begin{bmatrix} 52.025 \\ -4.783 \end{bmatrix}, \begin{bmatrix} 38.152 \\ 19.542 \end{bmatrix}, \begin{bmatrix} 19.269 \\ 53.272 \end{bmatrix}, \right. \\
&\qquad \left. \begin{bmatrix} 21.855 \\ 43.64 \end{bmatrix}, \begin{bmatrix} 37.592 \\ 39.889 \end{bmatrix}, \begin{bmatrix} 13.213 \\ 4.272 \end{bmatrix}, \begin{bmatrix} -6 \\ 2 \end{bmatrix} \right\}; \qquad (4.46)
\end{aligned}$$

$$\begin{aligned}
\mathbf{X}_2(t_1) &= \left\{ \begin{bmatrix} -0.697 \\ -36.452 \end{bmatrix}, \begin{bmatrix} -30.155 \\ -42.664 \end{bmatrix}, \begin{bmatrix} -2.152 \\ -42.812 \end{bmatrix}, \right. \\
&\qquad \left. \begin{bmatrix} 36.501 \\ -43.324 \end{bmatrix}, \begin{bmatrix} 26.866 \\ -40.747 \end{bmatrix} \right\}, \\
\mathbf{D}_2 &= \left\{ \begin{bmatrix} 31.917 \\ 17.622 \end{bmatrix}, \begin{bmatrix} 52.025 \\ -4.783 \end{bmatrix}, \begin{bmatrix} 38.152 \\ 19.542 \end{bmatrix}, \right. \\
&\qquad \left. \begin{bmatrix} 19.269 \\ 53.272 \end{bmatrix}, \begin{bmatrix} 21.855 \\ 43.64 \end{bmatrix} \right\}. \qquad (4.47)
\end{aligned}$$

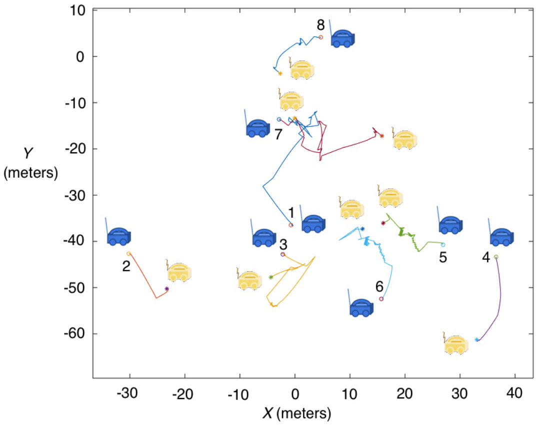


Figure 4.2: The trajectories of eight robots. The dark and the light robots are at initial and final positions, respectively.

It can be seen in Figures 4.2 and 4.4 that the two groups of robots are able to reach the predefined formation in Figures 4.3 and 4.5. Definition 4.3.2 and (4.13) allow the robots on the edge to map themselves to edge points in the predefined formation. However, the trajectories are not smooth, due to the frequent activation of Protocol 4.4.1. It can also be seen in Figure 5.6 from the many sudden jumps in the plot. During two neighbouring jumps, the curve is smooth. When the mapping of the robots changes, a jump can clearly be observed.

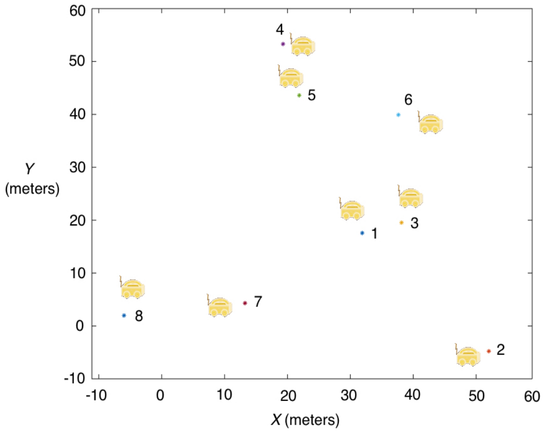


Figure 4.3: The predefined formation that includes eight points.

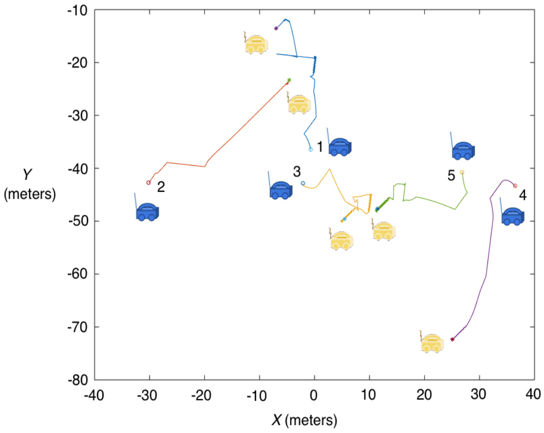


Figure 4.4: The trajectories of five robots. The dark and the light robots are at initial and final positions, respectively.

Moreover, it can be seen from Figure 5.6 that at around $t = 1.1$, 2.6, 4.3, 4.8, 5.8 seconds, the curve is very close to zero, and then, it jumps to a value much bigger than zero. This is because at these instants, the robots have reached their steady states under a mapping. However, this mapping is not a global optimum mapping and as **D** is justified in Section 4.4.2, the robots escape from this false mapping. Then, as explained in Section 4.4.3, a better mapping is found until the global optimal mapping is reached.

Note that Figure 4.7 shows the number of protocol activations and the simulation time under different probability settings. It can be seen that as the probability increases, the simulation time presents a ‘U’ shaped curve. This is due to the trade-off between the need for fast response and avoiding the trap of the infinite formation loop. In fact, we observe from the simulations that when the probability is smaller than 0.7, the robot team is always trapped in some infinite loop.

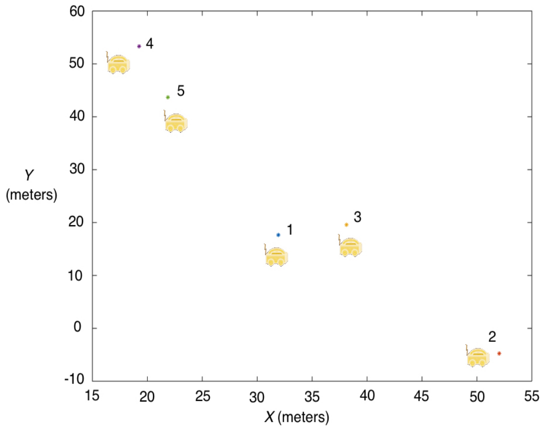


Figure 4.5: The predefined formation that includes five points.

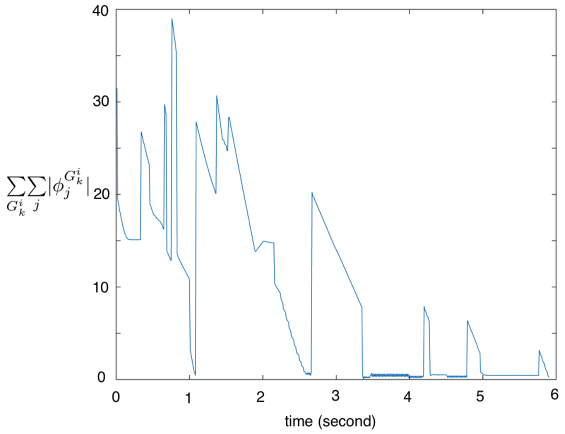


Figure 4.6: Plot of norm sum of control edges for a team of six robots with p = 0.94.

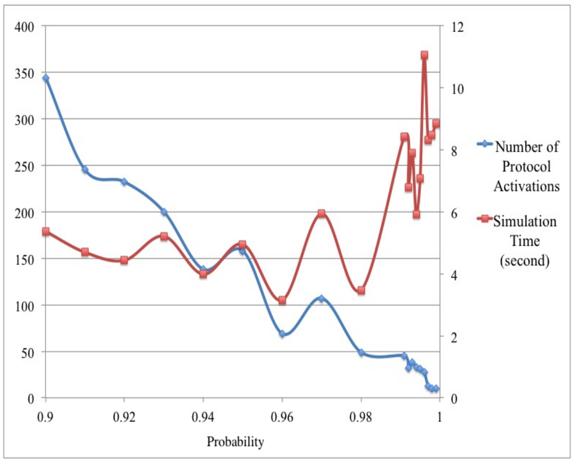


Figure 4.7: Different numbers of protocol activation and simulation time under different probability for a team of six robots.

4.6 Summary

Formation control for a team of robots was studied in this chapter. The limited sensing and unavailable communication made the robot team fully distributed in their information flow. The objective is to drive all the robots to the displacements from the predefined formation, which is available to all the robots. A novel controller and a coordination protocol were designed. To improve the performance of the controller and the protocol, a method was given to evaluate the predefined formation so that the formation imperfections can be detected and the global equilibrium is unique. The concept of minimal local topology was introduced to enhance the robustness and reduce the conservativeness in the connectivity maintaining condition. Moreover, probability was introduced to reduce the protocol-induced computation burden. The modified protocol is able to avoid the protocol-induced trap of an infinite formation loop.

Chapter 5

Leader-Follower Formation and Collision Avoidance

Distributed formation is studied for a team of mobile robots of leaders and followers. Followers are able to sense the relative displacements to neighbouring followers and all of the leaders, and the leaders can be sensed by the followers. Based on such assumption of sensing, distributed formation control scheme is designed, under which both groups of followers and leaders are fully independent with exchangeable roles within their own group. The leaders are free to have an arbitrary formation, and around the leaders, the followers need to reach a regular polygon formation with a suitable orientation. Distributed control laws and localised collision avoidance algorithms are designed for each follower, and they use only local displacements. Speed and acceleration sensors are avoided. As the leaders and the followers are independent and exchangeable, both robot teams are scalable and robust against member failures and system delays. Simulation and practical experiments are conducted to verify the performance.

5.1 Introduction

In this chapter, formation control is studied for a group of two-wheeled robots. We have designed a self-organising scheme with the capacity of scalability. Such distributed scheme adopts incremental speed control and it has been proved in experiments to be robust against system delay and static friction. Moreover, the algorithms are based on local sensing and therefore, they can drive the following robots to optimality through only local interaction.

The followers' control is distributed but the leaders' sensing is not necessarily decentralised. The position measurements of the robots could be obtained in two vision-based schemes, similarly to those in [51, 54]. The leaders could have pan-controlled cameras to observe the followers. Through some basic feature extraction, measurements were sent to the related following robots. As collision avoidance is considered and desired relative distances are required, additional range sensors are needed. Alternatively, UAVs can be used to collect measurements of both angles and distances, and all of the UAVs should have a view of all of the leaders and the related followers. The individual robot has two wheels, whose speeds can be directly manipulated in an incremental manner to combat static friction and system delay. Only two other local robots are used to produce the control force, unlike those in [53, 59, 60] where some global information was used.

Adaptive desired positions for the individual robots are derived based on the displacements among neighbouring robots, and control laws are designed based on these positions. As a result, speed sensors and differential operations on positions are avoided, and this contributes to the robustness of the formation. There are static friction between the wheels and the ground, under which the control laws still have satisfactory performances. As the incremental controller takes uncertain amount of time to reach the suitable wheel speeds, the static friction leads to uncertain time delays, which means that the control law is robust to unknown delays and disturbances. The followers share the same controllers and function similarly and independently. The leaders are equally important and independent.

The most suitable objective formation is obtained by the algorithm based on view angles and relative distances, similar to those in [13, 47, 48, 50, 62]. Moreover, such formation changes whenever any leader moves or there is some disturbance among the followers. The control law is able to adapt to a varying (large or small) number of leaders and followers, as long as the followers can sense their neighbours and the leaders can be sensed by the followers. This makes the robot team scalable for both the leaders and followers. The followers can function with at least two followers and one leader, which means that the team is naturally robust against the failures of the leaders and the followers.

5.2 Desired Position

There is a team X of N two-wheel robots labeled as followers. Another group Z of M target robots are labeled as leaders. The leaders can be sensed by the followers, and the objective is for the followers to surround the leaders in an adaptive formation. To design the controller, some variables are defined. The orientation vector set E_X of the robot group X denotes the set of the orientations for group members of X. The sets V_l^X, V_r^X and V^X of the robot group X denote the speeds of the left and right wheels and the robot centres for group members of X. The sets of X, Z, E_X, V_l^X, V_r^X and V^X are defined below.

$$
\begin{aligned}
X &= \left\{ \begin{array}{cccc} x_1 & x_2 & \dots & x_N \end{array} \right\}, \\
Z &= \left\{ \begin{array}{cccc} z_1 & z_2 & \dots & z_M \end{array} \right\}, \\
E_X &= \left\{ \begin{array}{cccc} e_{x_1} & e_{x_2} & \dots & e_{x_N} \end{array} \right\}, \\
V_l^X &= \left\{ \begin{array}{cccc} v_l^{x_1} & v_l^{x_2} & \dots & v_l^{x_N} \end{array} \right\}, \\
V_r^X &= \left\{ \begin{array}{cccc} v_r^{x_1} & v_r^{x_2} & \dots & v_r^{x_N} \end{array} \right\}, \\
V^X &= \left\{ \begin{array}{cccc} v^{x_1} & v^{x_2} & \dots & v^{x_N} \end{array} \right\},
\end{aligned}
$$

where i) $x' \in \mathbf{R}^2$ for any $x' \in X$; ii) $z' \in \mathbf{R}^2$ for any $z' \in Z$; iii) $e_{x'} \in E_X$, $e_{x'} \in \mathbf{R}^2$ and $|e_{x'}| = 1$ for any $x' \in X$; iv) $v_l^{x'} \in V_l^X$ and $v_l^{x'} \in \mathbf{R}$ for any $x' \in X$; v) $v_r^{x'} \in V_r^X$ and $v_r^{x'} \in \mathbf{R}$ for any $x' \in X$; vi) $v^{x'} \in V^X$ and $v^{x'} \in \mathbf{R}$ for any $x' \in X$.

Practical robots have sizes, and we assume the robots as round with radii. Specifically, for robot $x' \in X$, its radius is $R_{x'}$. Thus, the sets R_X and R_Z of radii for the robot teams X and Z are below.

$$
\begin{aligned}
R_X &= \left\{ \begin{array}{cccc} R_{x_1} & R_{x_2} & \dots & R_{x_N} \end{array} \right\}, \\
R_Z &= \left\{ \begin{array}{cccc} R_{z_1} & R_{z_2} & \dots & R_{z_M} \end{array} \right\}.
\end{aligned}
$$

For practical robots, they should not collide with each other and we assume that there is an avoidance range R' among them. It means that for robots

x' and x'', robot x' considers collision avoidance if and only if $|x' - x''| \leq R'$. For robot $x_i \in X$, it can produce a desired position κ^{x_i} for $z_j \in Z$. The robots should also have an corner range R among them. It means that $|x_i - \kappa^{x_i}| = R$.

We derive the desired position κ^{x_i} for robot $x_i \in X$. We need the vectors to either the left or the right of the robots' orientations. For follower x_i and leader z_j, there exists a constant $\mathcal{A}$ such that

$$\begin{aligned} &(x_i - z_j)^T \mathcal{A} (x_i - z_j) = 0, \\ &|\mathcal{A} (x_i - z_j)| = |(x_i - z_j)| \,. \end{aligned}$$

Here, $\mathcal{A} (x_i - z_j)$ points to the right side of $(x_i - z_j)$.

Based on $\mathcal{A}$ and X, we have the reference set

$$W_{z_j}^{x_i} = \left[\begin{array}{cccc} s_{x_1}^{x_i,z_j} y_{x_1}^{x_i,z_j} & s_{x_2}^{x_i,z_j} y_{x_2}^{x_i,z_j} & \ldots & s_{x_N}^{x_i,z_j} y_{x_N}^{x_i,z_j} \end{array} \right],$$

where

$$\begin{aligned} y_{x_k}^{x_i,z_j} &= \operatorname{acos}\left(\frac{(x_i - z_j)^T (x_k - z_j)}{|x_i - z_j| \cdot |x_k - z_j|} \right), \\ s_{x_k}^{x_i,z_j} &= \operatorname{sign}((x_k - z_j)^T \mathcal{A} (x_i - z_j)). \end{aligned}$$

For any $a \in \mathbf{R}$, sign$(a) = 1$ if $a \geq 0$; sign$(a) = -1$ if $a < 0$. Thus, we have

$$\begin{aligned} x_l^{x_i,z_j} &= \arg\min W_{z_j}^{x_i}, \\ x_r^{x_i,z_j} &= \arg\max W_{z_j}^{x_i}. \end{aligned} \tag{5.1}$$

For robot x_i, we have the desired position $\kappa_{z_j}^{x_i}$ from z_j below.

$$\kappa^{x_i} = \bar{z} + R \frac{\frac{x_l^{x_i,\bar{z}} - \bar{z}}{|x_l^{x_i,\bar{z}} - \bar{z}|} + \frac{x_r^{x_i,\bar{z}} - \bar{z}}{|x_r^{x_i,\bar{z}} - \bar{z}|}}{\left| \frac{x_l^{x_i,\bar{z}} - \bar{z}}{|x_l^{x_i,\bar{z}} - \bar{z}|} + \frac{x_r^{x_i,\bar{z}} - \bar{z}}{|x_r^{x_i,\bar{z}} - \bar{z}|} \right|}, \tag{5.2}$$

where

$$\bar{z} = \frac{1}{M}\sum_{z_j \in Z} z_j.$$

It can be seen from (5.2) that the positions of the leaders are averaged to derive the desired position for robot x_i. The followers' desired positions could reach a regular polygon around $\bar{z}$ with a suitable orientation. However, that may not be the most suitable way to corner a group of leaders. Alternatively, we can have (5.3) as an alternative for (5.2).

$$\kappa^{x_i} = \frac{1}{M}\sum_{z_j \in Z} \kappa^{x_i}_{z_j}, \tag{5.3}$$

where

$$\kappa^{x_i}_{z_j} = z_j + R\frac{\frac{x_l^{x_i,z_j} - z_j}{\left|x_l^{x_i,z_j} - z_j\right|} + \frac{x_r^{x_i,z_j} - z_j}{\left|x_r^{x_i,z_j} - z_j\right|}}{\left|\frac{x_l^{x_i,z_j} - z_j}{\left|x_l^{x_i,z_j} - z_j\right|} + \frac{x_r^{x_i,z_j} - z_j}{\left|x_r^{x_i,z_j} - z_j\right|}\right|}.$$

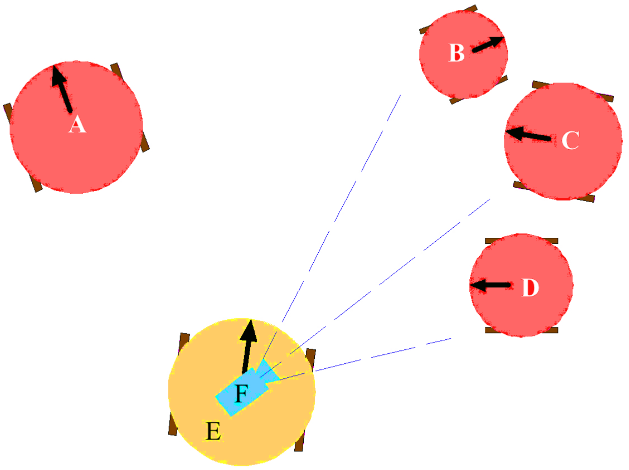


Figure 5.1: Four robots cornering robot E. Robot E has an onboard camera F.

Note that the followers' control is distributed since only neighbouring followers are needed in the controller design. Namely, for robot x_i, only two related followers are used. To derive κ^{x_i}, two possible scenarios can obtain enough information for the purpose. (Scenario A) For a group of N followers, N UAVs with surveillance cameras could be used to provide information for each followers. Each of them should have a view of all the leaders and three related followers suggested by (5.1). (Scenario B) Each leader could carry a pan-camera, whose field of view should be large enough to cover any of the three related followers. As a scenario example, consider Figure 5.1. Robot E is a leader and camera F is its pan-camera. F can rotate and capture the target view of any three related followers, such as robots B, C and D. The dynamics of the UAVs and the pan-cameras are omitted.

5.3 Model Description

To design the controller for each robot, the mathematical descriptions is derived as follow. Consider Figure 5.2. Assume that from instant k to $k+1$, robot A moves from A to A' along the trajectory $\overset{\frown}{AA'A''}$ (shown in Figure 5.2). As the control action is taken only at instants k and $k+1$, and the speeds change, if and only if a control action is taken, the speeds of robot A's wheels from k to $k+1$ are constant. As the robot has two wheels and a constant speed, the trajectories of robot A over different given intervals between k and $k+1$ fall on a circle with a center O and a radius $\left|\vec{OA}\right|$ (the vector from point O to point A is $\vec{OA} \in \mathbf{R}^2$ and $\left|\vec{OA}\right|$ is the distance between the two points). It also means that during a given interval, the trajectories of A_l, A_r and A are on concentric circles.

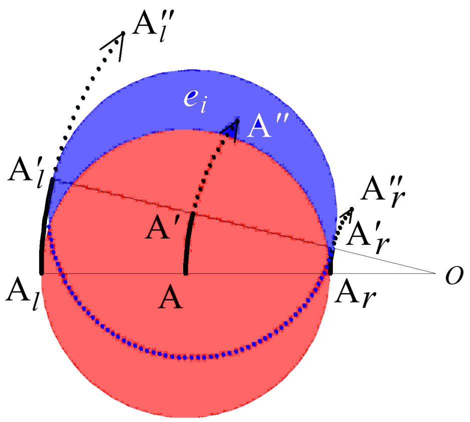


Figure 5.2: A single robot moves over a small displacement.

To test the algorithms in experiments, we used Arduino robots that are built with step motors and have speed modes, at which the robot wheel speeds can be directly manipulated. Therefore, we only consider the wheel speeds and their incremental changes as the only variables in the robot model. Assume that for robot x_i, the speeds of the left and right wheels are $v_l^{x_i}(k)$ and $v_r^{x_i}(k)$, respectively. If $v_l^{x_i}(k) > 0$, then the left wheel is rotating clockwise; if $v_r^{x_i}(k) > 0$, then the right wheel is rotating clockwise. If $v_l^{x_i}(k) < 0$, then the left wheel is rotating anti-clockwise; if $v_r^{x_i}(k) < 0$, then the right wheel is rotating anti-clockwise. If $v_l^{x_i}(k) = 0$, then the left wheel stops rotating; if $v_r^{x_i}(k) = 0$, then the right wheel stops rotating. With $v_l^{x_i}(k)$ and $v_r^{x_i}(k)$, we can derive the mathematical model to describe the dynamics of a robot.

For robot x_i, we have the wheels' shift velocity below.

$$\vec{v}_l^{x_i}(k) = v_l^{x_i}(k)e_{x_i}, \;\; \vec{v}_r^{x_i}(k) = v_r^{x_i}(k)e_{x_i}.$$

Robot x_i is regarded as round, whose centre's speed is $v^{x_i}(k)$. It can be seen that

$$\vec{v}^{x_i}(k) = \frac{\vec{v}_l^{x_i}(k) + \vec{v}_r^{x_i}(k)}{2} = v^{x_i}(k)e_{x_i}.$$

It follows that $v^{x_i}(k) = \frac{v_l^{x_i}(k)+v_r^{x_i}(k)}{2}$. The trajectories of A_l, A and A_r fall on concentric circles, whose radiuses are $r_l^{x_i}(k)$, $r^{x_i}(k)$ and $r_r^{x_i}(k)$ below.

$$\begin{aligned}
r_l^{x_i}(k) &= 2R + 2R \cdot \operatorname{sign}\left(\left(\vec{v}_l^{x_i}\right)^T \vec{v}_r^{x_i}\right) \cdot \operatorname{sign}\left(\left(\vec{v}_l^{x_i}\right)^T \left(\vec{v}_l^{x_i} - \vec{v}_r^{x_i}\right)\right) \frac{|\vec{v}_r^{x_i}|}{|\vec{v}_l^{x_i} - \vec{v}_r^{x_i}|}, \\
r_r^{x_i}(k) &= 2R + 2R \cdot \operatorname{sign}\left(\left(\vec{v}_l^{x_i}\right)^T \vec{v}_r^{x_i}\right) \cdot \operatorname{sign}\left(\left(\vec{v}_r^{x_i}\right)^T \left(\vec{v}_r^{x_i} - \vec{v}_l^{x_i}\right)\right) \frac{|\vec{v}_l^{x_i}|}{|\vec{v}_l^{x_i} - \vec{v}_r^{x_i}|}, \\
r^{x_i}(k) &= \frac{r_l^{x_i}(k) + r_r^{x_i}(k)}{2}.
\end{aligned}$$

During a sampling period τ, we have a small angular displacement $\theta^{x_i}(k)$ below.

$$\begin{aligned}
\theta^{x_i}(k) = {} & \operatorname{sign}\left(v_l^{x_i}\right) \cdot \operatorname{sign}\left(\left(\vec{v}_l^{x_i}\right)^T \left(\vec{v}_l^{x_i} - \vec{v}_r^{x_i}\right)\right) \\
& \cdot \operatorname{sign}\left(\left(\vec{v}_l^{x_i}\right)^T \left(\vec{v}_l^{x_i} + \vec{v}_r^{x_i}\right)\right) \tau \frac{|v^{x_i}(k)|}{r^{x_i}(k)}.
\end{aligned}$$

Based on $\theta^{x_i}(k)$, we can update robot x_i's orientation $e_{x_i}(k)$ below.

$$e_{x_i}(k+1) = \begin{bmatrix} \cos(\theta^{x_i}(k)) & -\sin(\theta^{x_i}(k)) \\ \sin(\theta^{x_i}(k)) & \cos(\theta^{x_i}(k)) \end{bmatrix} e_{x_i}(k).$$

To sum up, we use the equation below to describe robot x_i's dynamic.

$$x_i(k+1) = x_i(k) + 2r^{x_i}(k)\sin(\frac{\theta^{x_i}(k)}{2}) \cdot \begin{bmatrix} \cos(\frac{\theta^{x_i}(k)}{2}) & -\sin(\frac{\theta^{x_i}(k)}{2}) \\ \sin(\frac{\theta^{x_i}(k)}{2}) & \cos(\frac{\theta^{x_i}(k)}{2}) \end{bmatrix} e_{x_i}(k).$$

As we can directly manipulate the speeds of the wheels in an incremental manner, the robot's control inputs for robot x_i take the form below.

$$\begin{aligned} v_l^{x_i}(k+1) &= v_l^{x_i}(k) + f_l^{x_i}(k), \\ v_r^{x_i}(k+1) &= v_r^{x_i}(k) + f_r^{x_i}(k). \end{aligned}$$

We then need to find out the appropriate control forces for $f_l^{x_i}(k)$ and $f_r^{x_i}(k)$, which takes account of the desired speeds, wheel balance and collision avoidance.

5.4 Desired Speed

The desired speed is the robot's speed, under which the control force is zero and the robot reach its desired position in uniform circular motion. In following sections, the controller design aims to transform the current robot speed to its desired speed.

Assume in Figure 5.3 that robot A moves to A' at its desired speed. The radius $r^d(k)$ of its robot centre's trajectory $\overset{\frown}{AA'}$ is below.

$$r^d(k) = \frac{|\vec{AA'}|}{2\sqrt{1 - \left(\frac{e_i^T(k)\vec{AA'}}{|\vec{AA'}|}\right)^2}}, \tag{5.4}$$

Based on $r^d(k)$, we can derive the radiuses $r_l^d(k)$ and $r_r^d(k)$ of the left and right wheels' trajectories $\overset{\frown}{A_l A'_l}$ and $\overset{\frown}{A_r A'_r}$ below.

$$\begin{aligned} r_l^d(k) &= r^{x_i}(k) + R \cdot \mathrm{sign}\left(e_i^T(k)\mathcal{A}\vec{AA'}\right), \\ r_r^d(k) &= r^{x_i}(k) - R \cdot \mathrm{sign}\left(e_i^T(k)\mathcal{A}\vec{AA'}\right). \end{aligned} \tag{5.5}$$

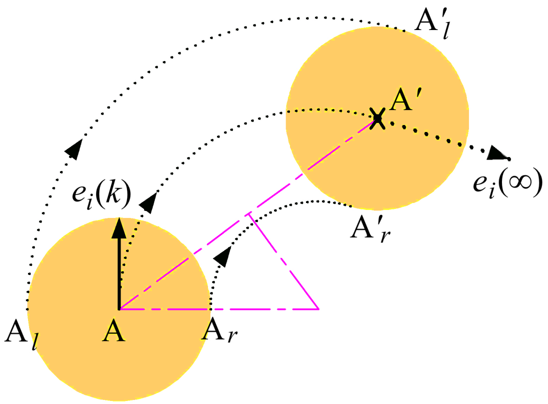


Figure 5.3: Robot A reaches its desired position under the desired speed.

Thus, we can find the desired speeds through the equations below.

$$\begin{aligned} &\frac{|\vec{AA'}|}{2\sqrt{1-\left(\frac{e_i^T(k)\vec{AA'}}{|\vec{AA'}|}\right)^2}} \\ &= 2R + R \cdot \mathrm{sign}\left(\left(\vec{v}_l^{x_i}\right)^T \vec{v}_r^{x_i}\right) \cdot \mathrm{sign}\left(\left(\vec{v}_l^{x_i}\right)^T \left(\vec{v}_l^{x_i} - \vec{v}_r^{x_i}\right)\right) \frac{|\vec{v}_l^{x_i}| + |\vec{v}_r^{x_i}|}{|\vec{v}_l^{x_i} - \vec{v}_r^{x_i}|}, \\ &\quad r^{x_i}(k) + R \cdot \mathrm{sign}\left(e_i^T(k)\mathcal{A}\vec{AA'}\right) \\ &= 2R + 2R \cdot \mathrm{sign}\left(\left(\vec{v}_l^{x_i}\right)^T \vec{v}_r^{x_i}\right) \cdot \mathrm{sign}\left(\left(\vec{v}_l^{x_i}\right)^T \left(\vec{v}_l^{x_i} - \vec{v}_r^{x_i}\right)\right) \frac{|\vec{v}_r^{x_i}|}{|\vec{v}_l^{x_i} - \vec{v}_r^{x_i}|}, \\ &\quad r^{x_i}(k) - R \cdot \mathrm{sign}\left(e_i^T(k)\mathcal{A}\vec{AA'}\right) \\ &= 2R + 2R \cdot \mathrm{sign}\left(\left(\vec{v}_l^{x_i}\right)^T \vec{v}_r^{x_i}\right) \\ &\quad \cdot \mathrm{sign}\left(\left(\vec{v}_r^{x_i}\right)^T \left(\vec{v}_r^{x_i} - \vec{v}_l^{x_i}\right)\right) \frac{|\vec{v}_l^{x_i}|}{|\vec{v}_l^{x_i} - \vec{v}_r^{x_i}|}. \end{aligned} \tag{5.6}$$

It is convenient to use the desired and current speeds to design the controllers. But deriving the solution of (5.6) needs complex computation. It can be seen that $r_l^d(k)$, $r_r^d(k)$ and $r^d(k)$ are readily obtainable by (5.4) and (5.5). Below, we design controllers based on $r_l^d(k)$, $r_r^d(k)$ and $r^d(k)$.

5.5 Controller Design

In the previous section, desired position and speed are derived for each robot. In this section, the controller is proposed to transform the current speed to the desired speed such that the robot can reach its desired position. The only available information is the orientation and relative displacement, and no differential operation is used.

Control of Center Speed

It can be seen that for robot x_i, the larger $r^{x_i}(k)$ is, the less it needs to rotate to point to its desired position. When $r^{x_i}(k)$ is infinitely large, the robot's orientation directly points to its desired position. Thus, it is reasonable to expect that under the control force, $r^{x_i}(k)$ gets larger and larger. We then have the control force below.

$$\begin{aligned} f_l^c(k) &= \alpha_l \frac{r_l^{x_i}(k)}{r_r^{x_i}(k)} \left(\frac{r_l^{x_i}(k-1)}{r_l^{x_i}(k)} - \sigma \right) \cdot \operatorname{sign}\left(e_i^T(k)(x_i(k) - \kappa^{x_i(k)}) \right), \\ f_r^c(k) &= \alpha_r \frac{r_r^{x_i}(k)}{r_l^{x_i}(k)} \left(\frac{r_r^{x_i}(k-1)}{r_r^{x_i}(k)} - \sigma \right) \cdot \operatorname{sign}\left(e_i^T(k)(x_i(k) - \kappa^{x_i(k)}) \right), \quad (5.7) \end{aligned}$$

where $\alpha_l, \alpha_r > 0$ and $\sigma < 1$ are to be specified.

Note that when $f_l^c(k) = f_l^c(k) = 0$, we have $\frac{r_l^{x_i}(k-1)}{r_l^{x_i}(k)} = \sigma$ and $\frac{r_r^{x_i}(k-1)}{r_r^{x_i}(k)} = \sigma$, which means that both radiuses are enlarging. $\frac{r_l^{x_i}(k)}{r_r^{x_i}(k)}$ and $\frac{r_r^{x_i}(k)}{r_l^{x_i}(k)}$ are the speed ratio between the left and right wheels; α_l and α_r denote the voltage-speed ratio difference between the left and right wheels.

Balance Control of Wheel Speeds

At the fixed speeds, the trajectories of robots fall on fixed circles. For robot A in Figure 5.4, we can derive its desired position $A_2'' = \kappa^A$ by (5.2). It is our consideration that at robot A's desired speed, A_2'' falls on robot A's trajectory by $r^A(k) = |AO_2|$, which means that the trajectory $\overset{\frown}{AA_2'A_2''}$ is from A to A_2''.

For robot A, the speeds of the two wheels are balanced if $r^A(k) = |AO_2|$. However, the speeds are possibly unbalanced:

i) either the left wheel is under-actuated if
the robot's trajectory is $A\overset{\frown}{A_1'}A_1''$ and $r^A(k) = |AO_1|$, or

ii) the right wheel is under-actuated
if the robot's trajectory is $A\overset{\frown}{A_3'}A_3''$ and $r^A(k) = |AO_3|$.

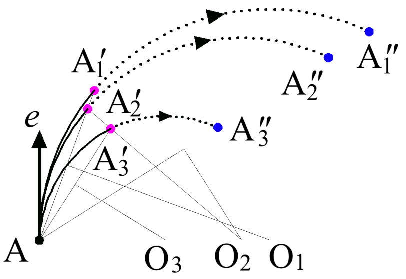


Figure 5.4: Robot A has overshot, undershot and a desired shot towards its desired position at different speeds. are centres of the trajectories .

For robot x_i, we can also derive the radius $r^d(k)$ at the desired speed. It can be seen that i) the left wheel is under-actuated if $r^{x_i}(k) < r^d(k)$, and ii) the right wheel is under-actuated if $r^{x_i}(k) > r^d(k)$. To sum up, we have the following correction based on the controller in (5.7).

$$
\begin{aligned}
\delta &= \epsilon \left(\frac{r^{x_i}(k)}{r^{x_i}(k-1)} - \sigma \right), \\
f_l^b(k) &= f_l^c(k) + \operatorname{sign}\left((\kappa^{x_i} - x_i(k))^T \mathcal{A} e_{x_i}(k) \right) \operatorname{sign}\left((\kappa^{x_i} - x_i(k))^T e_{x_i}(k) \right) \delta, \\
f_r^b(k) &= f_r^c(k) - \operatorname{sign}\left((\kappa^{x_i} - x_i(k))^T \mathcal{A} e_{x_i}(k) \right) \operatorname{sign}\left((\kappa^{x_i} - x_i(k))^T e_{x_i}(k) \right) \delta,
\end{aligned}
$$

where ϵ is a gain to be determined.

Collision Avoidance

The robots are required not to collide with other robots and not to trespass beyond borders. In this section, a collision avoidance is designed based on such limited information under an avoidance range R'. Assume that there are four robots A, B, C and D, and a border E, whose inter-robot and robot-border distances are smaller than R'. The radii of robot A and D are R_A and R_D. If any of the distances from A, B, C and E to D is smaller than R', then robot D needs to enter the collision avoidance mode. For robot D, in order to avoid collision with robot A, $e_D(k)$ should not be in the blue area, which is defined by $D\vec{A}_r$ and $D\vec{A}_l$. It follows that

$$\begin{aligned} D\vec{A}_r &= \begin{bmatrix} \cos\frac{\angle A_r D A_l}{2} & -\sin\frac{\angle A_r D A_l}{2} \\ \sin\frac{\angle A_r D A_l}{2} & \cos\frac{\angle A_r D A_l}{2} \end{bmatrix} D\vec{A}, \\ D\vec{A}_l &= \begin{bmatrix} \cos(-\frac{\angle A_r D A_l}{2}) & -\sin(-\frac{\angle A_r D A_l}{2}) \\ \sin(-\frac{\angle A_r D A_l}{2}) & \cos(-\frac{\angle A_r D A_l}{2}) \end{bmatrix} D\vec{A}, \end{aligned}$$

where

$$\frac{\angle A_r D A_l}{2} = \operatorname{asin}(\frac{R_A + R_D}{\left|D\vec{A}\right|}).$$

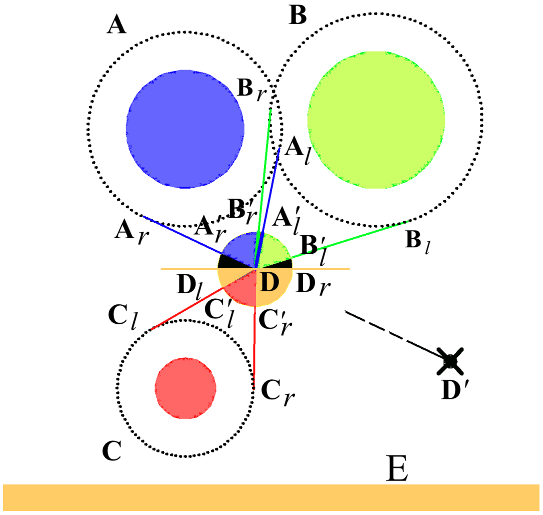


Figure 5.5: Robot D with collision avoidance and boarder restriction.

In theory with $e_D(k+1)$ obtainable,

i) if either $e_D^T(k)\mathcal{A}A_r\vec{D} \cdot e_D^T(k)\mathcal{A}A_l\vec{D} \leq 0$, $\left(\vec{v}^D(k)\right)^T A_r\vec{D} \geq 0$, and $\left(\vec{v}^D(k)\right)^T A_l\vec{D} \geq 0$, or $e_D^T(k+1)\mathcal{A}A_r\vec{D} \cdot e_D^T(k+1)\mathcal{A}A_l\vec{D} \leq 0$, $\left(\vec{v}^D\right)^T (k+1)A_r\vec{D} \geq 0$, and $\left(\vec{v}^D\right)^T (k+1)A_l\vec{D} \geq 0$, then we have

$$\gamma_A^D = \{\ A_r\vec{D} \quad A_l\vec{D}\ \};$$

ii) if otherwise, then robot D does not collide with robot A, and we have

$$\gamma_A^D = \emptyset;$$

iii) $\gamma_A^A = \emptyset$.

In practical with $e_D(k+1)$ unobtainable,

i) if $e_D^T(k)\mathcal{A}\vec{A_rD} \cdot e_D^T(k)\mathcal{A}\vec{A_lD} \leq 0$, $\left(\vec{v}^D(k)\right)^T \vec{A_rD} \geq 0$, and $\left(\vec{v}^D(k)\right)^T \vec{A_lD} \geq 0$, then we have the avoidance set for A as

$$\gamma_A^D = \left\{ \begin{array}{cc} \vec{A_rD} & \vec{A_lD} \end{array} \right\};$$

ii) if otherwise, then robot D does not collide with robot A, and we have

$$\gamma_A^D = \emptyset;$$

iii) $\gamma_A^A = \emptyset$.

The same is between robot D and B, between robot D and C, between robot D and boarder E, given that any of their relative distances is smaller than R'. To avoid collisions, $e_D(k)$ should also not be in the green, red and yellow areas, which are defined by $\vec{DB_r}$ and $\vec{DB_l}$, $\vec{DC_r}$ and $\vec{DC_l}$, and $\vec{DE_r}$ and $\vec{DE_l}$, respectively.

Similarly, for robot x_i and based on the robots from X and Z that are inside the avoidance range R' of robot x_i, we have the avoidance set γ^{x_i} as follows

$$\gamma^{x_i} = \left\{ \begin{array}{cccccccc} \gamma_{x_1}^{x_i} & \gamma_{x_2}^{x_i} & \ldots & \gamma_{x_N}^{x_i} & \gamma_{z_1}^{x_i} & \gamma_{z_2}^{x_i} & \ldots & \gamma_{z_M}^{x_i} \end{array} \right\}.$$

Based on γ^{x_i} and κ_{x_i}, we have

$$\gamma^* = \arg\max_{\gamma' \in \gamma^{x_i}} \frac{\left(\kappa_{x_i} - x_i\right)^T \gamma'}{|\left(\kappa_{x_i} - x_i\right)|}.$$

iv) If $\gamma^{x_i} = \emptyset$ and robot x_i is in the collision avoidance mode, then robot x_i enters the normal mode, and

$$v_l^{x_i}(k+1) = 0,\ \ v_r^{x_i}(k+1) = 0;$$

v) if $\gamma^{x_i} = \emptyset$ and robot x_i is in the normal mode,
then continue to the next iteration;

vi) if $\gamma^{x_i} \neq \emptyset$ and robot x_i is in the normal mode,
then robot x_i enters the collision avoidance mode, and

$$v_l^{x_i}(k+1) = 0,\ \ v_r^{x_i}(k+1) = 0;$$

vii) if $\gamma^{x_i} \neq \emptyset$ and robot x_i is already in the collision avoidance mode,
then we have

$$v_l^{x_i}(k+1) = v_l^{x_i}(k) + \eta_l \mathrm{sign}\left((\kappa_{x_i} - x_i)^T \mathcal{A}\gamma^*\right) \cdot \mathrm{acos}\left(\frac{(\kappa_{x_i} - x_i)^T \gamma^*}{|\,(\kappa_{x_i} - x_i)\,|}\right),$$

$$v_r^{x_i}(k+1) = v_r^{x_i}(k) - \eta_r \mathrm{sign}\left((\kappa_{x_i} - x_i)^T \mathcal{A}\gamma^*\right) \cdot \mathrm{acos}\left(\frac{(\kappa_{x_i} - x_i)^T \gamma^*}{|\,(\kappa_{x_i} - x_i)\,|}\right),$$

where $\eta_l > 0$ and $\eta_r > 0$ are the angle-to-wheel-speed gain to be determined.

5.6 System Analysis

In previous sections, we have designed the distributed control laws to drive the robots to their desired speeds. In this section, we analyze how robots can collectively reach their desired positions at their desired speeds. It is assumed that the control is unsynchronised among the robots. Therefore, at a given instant, only one robot is making control actions. Moreover, the speeds of robots are changed incrementally once the control action is taken.

It can be seen in both Figures 5.6 and 5.7 that at the desired speeds, robot x_2 moves towards κ^{x_2} along the trajectory $\overset{\frown}{x_2\kappa^{x_2}}$, during which the angle $\angle x_2Rz_1$ (the angle from the vector $\overrightarrow{Rx_2}$ to the vector $\overrightarrow{Rz_1}$) approaches zero.

$$\cos\angle x_2Rz_1 = \frac{|x_2R|^2 + |z_1R|^2 - |x_2z_1|^2}{2\,|x_2R| \cdot |z_1R|^2}. \tag{5.8}$$

By (5.9) and $\angle x_2Rz_1 \to 0$, we have

$$\begin{aligned} \frac{|x_2R|^2 + |z_1R|^2 - |x_2z_1|^2}{2\,|x_2R| \cdot |z_1R|^2} &= 1, \\ |x_2R - z_1R| &= |x_2z_1|\,. \end{aligned} \tag{5.9}$$

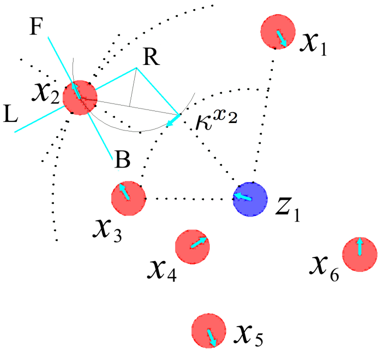


Figure 5.6: Robot x_2 moves backward towards its desired position κ^{x_2} at the desired speeds.

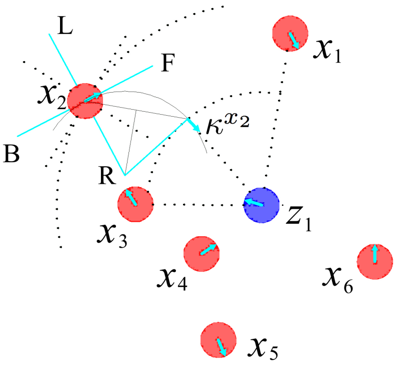


Figure 5.7: Robot x_2 moves forward towards its desired position κ^{x_2} at the desired speeds.

Similarly, we have

$$\cos\angle Rz_1x_2 = \frac{|Rz_1|^2 + |x_2z_1|^2 - |Rx_2|^2}{2\,|Rz_1| \cdot |x_2z_1|^2}. \tag{5.10}$$

Substitute (5.10) into (5.9), it follows that $\cos\angle Rz_1x_2 = 1$ and $\angle Rz_1x_2 = 0$.

For robot x_2, given $\vec{x_2F}^T \vec{x_2\kappa^{x_2}} < 0$ and $\vec{x_2F}^T \mathcal{A}\vec{x_2\kappa^{x_2}} > 0$, we have a non-increasing displacement error $\left|x_2(k) - \kappa^{x_2(k)}\right|$ of robot x_2 below if $v_l(k-1) < 0$ and $v_l(k-1) + v_r(k-1) < 0$.

$$\left|x_2(k) - \kappa^{x_2(k)}\right| \leq \left|x_2(k-1) - \kappa^{x_2(k-1)}\right|. \tag{5.11}$$

Similarly, given $\vec{x_2F}^T \vec{x_2\kappa^{x_2}} < 0$ and $\vec{x_2F}^T \mathcal{A}\vec{x_2\kappa^{x_2}} < 0$, if $v_r(k-1) < 0$ and $v_l(k-1) + v_r(k-1) < 0$, then we have (5.11);

Given $\vec{x_2F}^T \vec{x_2\kappa^{x_2}} > 0$ and $\vec{x_2F}^T \mathcal{A}\vec{x_2\kappa^{x_2}} > 0$, if $v_l(k-1) > 0$ and $v_l(k-1) + v_r(k-1) > 0$, then we have (5.11);

Given $\overrightarrow{x_2F}^T \overrightarrow{x_2\kappa^{x_2}} > 0$ and $\overrightarrow{x_2F}^T \mathcal{A}\overrightarrow{x_2\kappa^{x_2}} < 0$, if $v_r(k-1) > 0$ and $v_l(k-1) + v_r(k-1) > 0$, then we have (5.11).

Similarly, for robot x_i at its desired speed, its displacement error $|x_i(\infty) - \kappa^{x_i(\infty)}| \rightarrow 0$.

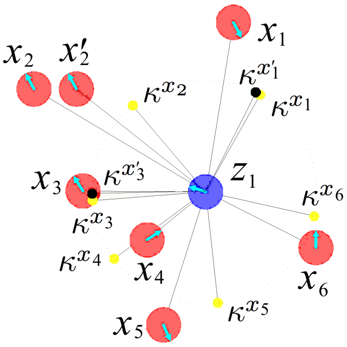


Figure 5.8: Change of the view angle errors after the movement of robot x_2.

It can be seen in Figure 5.8 that $x_l^{x_2,z_1} = x_1$ and $x_r^{x_2,z_1} = x_3$ for robot x_2. We denote the angle error of robot x_2 as $|\angle\kappa^{x_2}z_1x_3(k) - \angle\kappa^{x_2}z_1x_1(k)|$ at instant k. In Figure 5.8, robot x_2 moves to x_2' from instant k to $k+1$ if it follows that

$$\begin{gathered}(v_l(k)+v_r(k))\text{sign}\left(e^T(k)\left(x_i(k)-\kappa^{x_i(k)}\right)\right) > 0,\\ \left|\frac{v_l(k)}{v_r(k)}\right|^{\text{sign}\left(e^T(k)\mathcal{A}\left(x_i(k)-\kappa^{x_i(k)}\right)\right)} > 1.\end{gathered}$$

For robot x_2, given $\overrightarrow{x_2F}^T \mathcal{A}\overrightarrow{x_2\kappa^{x_2}} > 0$, if $v_l(k-1)+v_r(k-1) > 0$, then we have

$$\begin{aligned}&\frac{(x_2(k)-z_1(k))^T\left(\kappa^{x_2(k)}-z_1(k)\right)}{|x_2(k)-z_1(k)|\cdot|\kappa^{x_2(k)}-z_1(k)|}\\ \geq\ &\frac{(x_2(k-1)-z_1(k-1))^T\left(\kappa^{x_2(k-1)}-z_1(k-1)\right)}{|x_2(k-1)-z_1(k-1)|\cdot|\kappa^{x_2(k-1)}-z_1(k-1)|},\\ &\frac{(x_3(k)-z_1(k))^T\left(\kappa^{x_3(k)}-z_1(k)\right)}{|x_3(k)-z_1(k)|\cdot|\kappa^{x_3(k)}-z_1(k)|}\end{aligned}$$

$$
\begin{aligned}
&\geq \frac{\left(x_3(k-1)-z_1(k-1)\right)^T\left(\kappa^{x_3(k-1)}-z_1(k-1)\right)}{|x_3(k-1)-z_1(k-1)|\cdot|\kappa^{x_3(k-1)}-z_1(k-1)|}, \\
&\frac{\left(x_1(k)-z_1(k)\right)^T\left(\kappa^{x_1(k)}-z_1(k)\right)}{|x_1(k)-z_1(k)|\cdot|\kappa^{x_1(k)}-z_1(k)|} \\
&\geq \frac{\left(x_1(k-1)-z_1(k-1)\right)^T\left(\kappa^{x_1(k-1)}-z_1(k-1)\right)}{|x_1(k-1)-z_1(k-1)|\cdot|\kappa^{x_1(k-1)}-z_1(k-1)|}.
\end{aligned}
\tag{5.12}
$$

Similarly, given $\vec{x_2F}^T\mathcal{A}\vec{x_2\kappa}^{x_2}<0$, if $v_l(k-1)+v_r(k-1)<0$, then we have (5.12).

To sum up, for robot x_i, we have

$$
\begin{aligned}
&\frac{\left(x_i(k)-\bar{z}(k)\right)^T\left(\kappa^{x_i(k)}-\bar{z}(k)\right)}{|x_i(k)-\bar{z}(k)|\cdot|\kappa^{x_i(k)}-\bar{z}(k)|} \\
&\geq \frac{\left(x_i(k-1)-\bar{z}(k-1)\right)^T\left(\kappa^{x_i(k-1)}-\bar{z}(k-1)\right)}{|x_i(k-1)-\bar{z}(k-1)|\cdot|\kappa^{x_i(k-1)}-\bar{z}(k-1)|}, \\
&\frac{\left(x_r^{x_i,\bar{z}}(k)-\bar{z}(k)\right)^T\left(\kappa^{x_r^{x_i,\bar{z}}(k)}-\bar{z}(k)\right)}{\left|x_r^{x_i,\bar{z}}(k)-\bar{z}(k)\right|\cdot\left|\kappa^{x_r^{x_i,\bar{z}}(k)}-\bar{z}(k)\right|} \\
&\geq \frac{\left(x_r^{x_i,\bar{z}}(k-1)-\bar{z}(k-1)\right)^T\left(\kappa^{x_r^{x_i,\bar{z}}(k-1)}-\bar{z}(k-1)\right)}{\left|x_r^{x_i,\bar{z}}(k-1)-\bar{z}(k-1)\right|\cdot\left|\kappa^{x_r^{x_i,\bar{z}}(k-1)}-\bar{z}(k-1)\right|}, \\
&\frac{\left(x_l^{x_i,\bar{z}}(k)-\bar{z}(k)\right)^T\left(\kappa^{x_l^{x_i,\bar{z}}(k)}-\bar{z}(k)\right)}{\left|x_l^{x_i,\bar{z}}(k)-\bar{z}(k)\right|\cdot\left|\kappa^{x_l^{x_i,\bar{z}}(k)}-\bar{z}(k)\right|} \\
&\geq \frac{\left(x_l^{x_i,\bar{z}}(k-1)-\bar{z}(k-1)\right)^T\left(\kappa^{x_l^{x_i,\bar{z}}(k-1)}-\bar{z}(k-1)\right)}{\left|x_l^{x_i,\bar{z}}(k-1)-\bar{z}(k-1)\right|\cdot\left|\kappa^{x_l^{x_i,\bar{z}}(k-1)}-\bar{z}(k-1)\right|},
\end{aligned}
\tag{5.13}
$$

if it follows that

$$
(v_l(k)+v_r(k))\text{sign}\left(e^T(k)\mathcal{A}\left(x_i(k)-\kappa^{x_i(k)}\right)\right)>0. \tag{5.14}
$$

By (5.13), we have a non-increasing angle error for robot x_i by $|\angle\kappa^{x_2}z_1x_3(k)-\angle\kappa^{x_2}z_1x_1(k)|\leq|\angle\kappa^{x_2}z_1x_3(k-1)-\angle\kappa^{x_2}z_1x_1(k-1)|$. Therefore, as long as (5.14) stands, we have robot x_i's angle error $|\angle\kappa^{x_2}z_1x_3(\infty)-\angle\kappa^{x_2}z_1x_1(\infty)|\rightarrow 0$. If the displacement and angle errors are zero for all of the followers, then the objective formation is reached.

5.7 Simulation and Experiment

In this section, three simulation scenarios and one experiment are conducted to test the proposed controllers. Multiple robots are involved in each test. These robots are divided into a group of followers and a group of leaders. The objective is for the followers to corner the leaders by reaching an optimised formation in a regular polygon with arbitrary orientation. In these simulation and experiment set-ups, the leaders do not have any particular objective to meet and they are free to have an arbitrary formation. For simplicity, the leaders remain stationary after specifying their initial locations in the following simulations and experiment since the dynamic of the leaders is not the concern of this chapter. In our experiment, through visual feature extraction, follower sizes, directions and distances relative to the leaders were sent to the following robots.

5.7.1 Simulation Results

Three simulation examples are given to demonstrate the effectiveness in this section. Four robots are cornering one robot in Figure 5.9, five robots are cornering one robot in Figure 5.12, and six robots are cornering two robots in Figure 5.15. For robot x_i at its desired speed, its displacement error is $\left|x_i(k) - \kappa^{x_i(k)}\right|$ and its angle error is $|\angle\kappa^{x_2}z_1x_3(k) - \angle\kappa^{x_2}z_1x_1(k)|$.

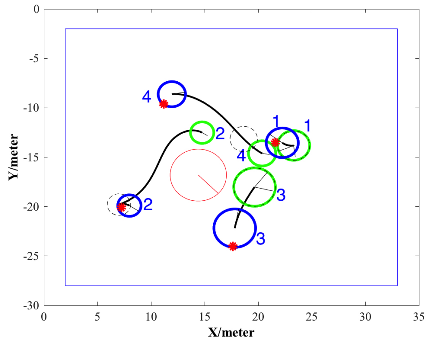


Figure 5.9: Trajectories of four robots cornering one robot (Scenario 1). The leader is red and the initial and final positions of the followers are green and blue.

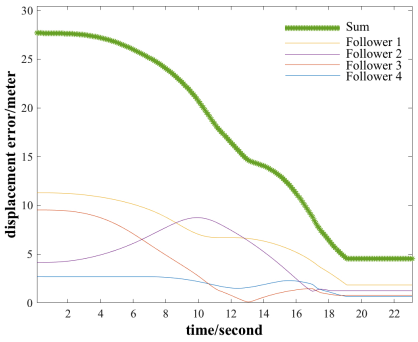


Figure 5.10: Displacement errors of four robots cornering one robot in Scenario 1.

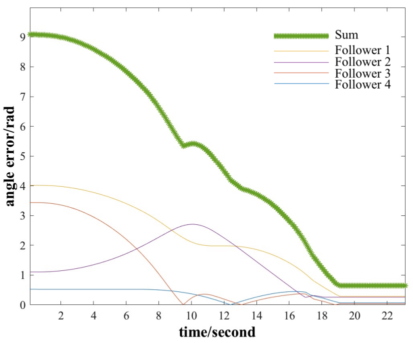


Figure 5.11: Angle errors of four robots cornering one robot in Scenario 1.

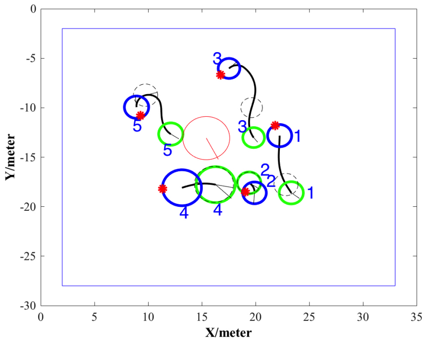


Figure 5.12: Trajectories of five robots cornering one robot (Scenario 2). The leader is red and the initial and final positions of the followers are green and blue.

Note that in Figures 5.9, 5.12 and 5.15, the stars are the ultimate desired positions for the followers. When the related followers are no more than their radius of size away from their desired positions, we consider that the robots have reached their desired positions. It can be seen that in Figure 5.9, to avoid collisions, robot 3 turned away from the leader, robot 1 turned away from robot 3. Robots 2 and 4 need not to avoid collisions and therefore, need no turning. As a result, robots 2 and 4 have longer trajectories than robots 1 and 3. Collision avoidance actions are also taken by robots 1 and 4 in Figure 5.12, and robots 1, 2 and 3 in Figure 5.15.

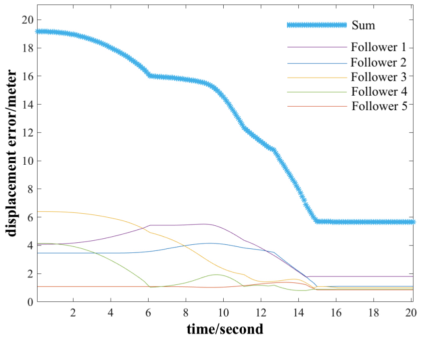


Figure 5.13: Displacement errors of five robots cornering one robot in Scenario 2.

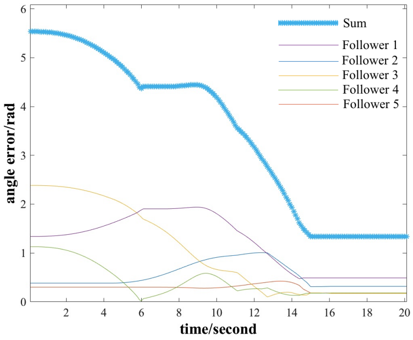


Figure 5.14: Angle errors of five robots cornering one robot Scenario 2.

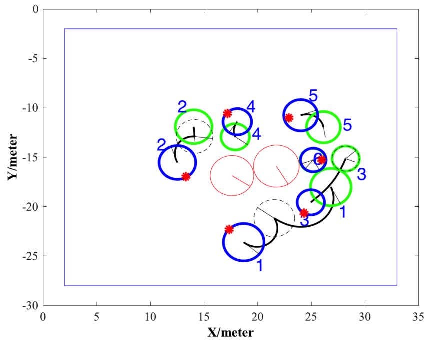


Figure 5.15: Trajectories of six robots cornering two robot (Scenario 3). The leader is red and the initial and final positions of the followers are green and blue.

Note that Figures 5.10, 5.13 and 5.16 are displacement errors and Figures 5.11, 5.14 and 5.17 are angle errors for Figure 5.9, 5.12 and 5.15, respectively. The errors in these figures all take absolute values, and the highest curves are the sums of the rest of the errors. If there are no collision avoidance actions, all of the curves are non-increasing. As robots brake to avoid collisions, the sums jump occasionally, which can be observed from all of the figures.

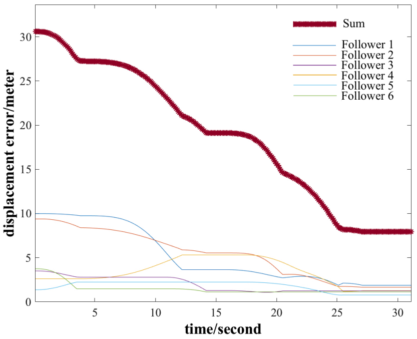


Figure 5.16: Displacement errors of six robots cornering two robots in Scenario 3.

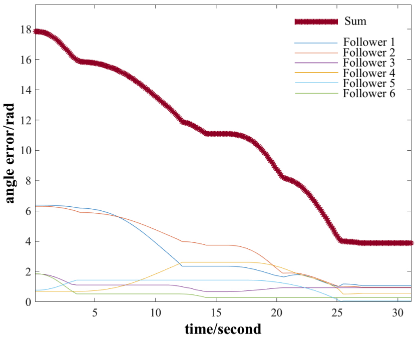


Figure 5.17: Angle errors of six robots cornering two robots in Scenario 3.

5.7.2 Experiment Results

Four Arduino robots are in a task to corner a robot. A pattern is drawn on the tops of all the robots to help extract sizes, center coordinates, and orientations of the robots. As they are placed on a small table and the area is limited, the robots need to take actions to avoid both collisions into each other and trespassing the boarders. The followers stop when their distances to their desired positions are less than their radiuses.

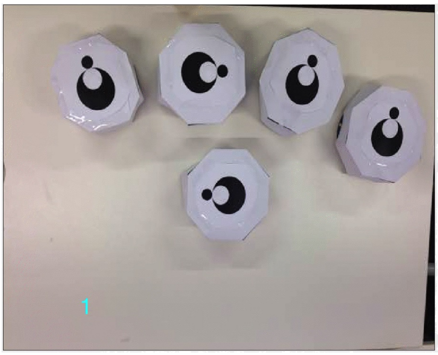


Figure 5.18: Snapshot of five Arduino robots in their initial positions.

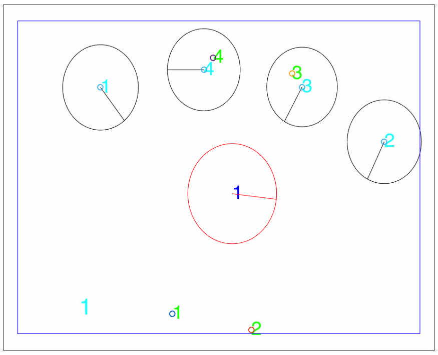


Figure 5.19: Key features extracted from the image of Figure 5.18, showing the five robots' initial positions.

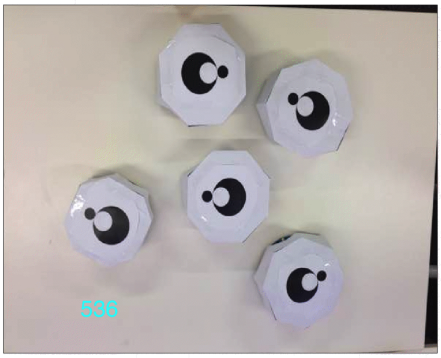


Figure 5.20: Snapshot of five Arduino robots in their final positions.

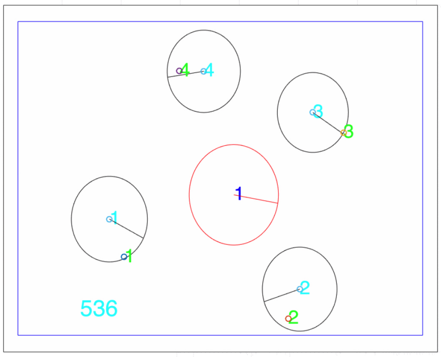


Figure 5.21: Key features extracted from the image of Figure 5.20.

The leader robot is static while the followers are trying to corner it. There are Wi-Fi shields on each robot, receiving signals from the local network. Based on these signals, the robots decide to either increase the wheel speeds or enter the collision avoidance mode. There is a wireless camera above the table, and images of the robots below are captured and sent to a computer base through the local network. Upon receiving these images, the computer extracts key features, based on which control signals are produced and ready to be sent to the local network. Similar with the setups for the simulation examples, there are static friction between the table and the wheels, and through the controllers, each robot graduate increases its motor input voltage and move.

As all the data communication are completed through the local network, there are limited delays on each data transmission.

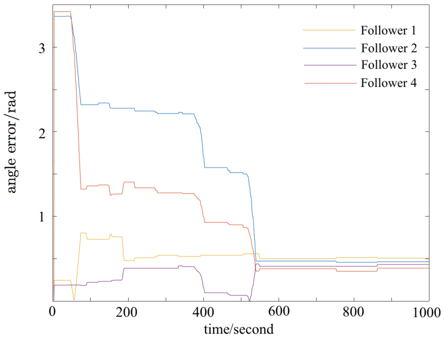


Figure 5.22: Angle errors of four Arduino robots cornering one robot in the experiment.

Note that Figures 5.18 and 5.20 show the initial and final positions of the robots, respectively. Figures 5.19 and 5.22 show the key features extracted from 5.18 and 5.22. Figure 5.22 shows the angle errors of the robots. It can be seen that due to collision and boarder avoidance, there are some obvious jumps on two curves and due to noises in the images, there are some small jumps on all of the curves. Ultimately, the robots are able to reach their desired positions.

5.8 Summary

Formation control was investigated in this chapter. The robots were two-wheeled, whose speeds could be directly manipulated by input voltages in an incremental manner. The objective is for the followers to corner the leaders in an adaptive formation. The followers are assumed to have limited sensing capacity and the leaders can be sensed. Possible setups to obtain the sensor measurements were briefly discussed, since they are not the focus of this chapter. Distributed algorithms were designed to derive the adaptive desired positions. Based on these positions, the control signals were generated, which were essentially produced from local displacements. No differential operations on positions were required, the needs for speed and acceleration sensors were eliminated, and this improved the robustness against static frictions and system delays.

Localised collision avoidance and border restriction were considered. Information of local robots and nearby borders were used to produce a unified set. Based on such local avoidance set, an optimisation scheme helped find the most suitable direction to escape the collisions. Simulation and experiment analyses showed that delays and disturbances would not affect the performance of the multi-robot system.

Chapter 6

Path Planning Among Obstacle Clusters

In previous chapters, formation control schemes have been proposed. Although one collision avoidance algorithm is designed, the schemes depend at least to some extent on the premise that the admissible space is smooth. In this chapter, we investigate path planning algorithm to drive robots through a congested area of multiple obstacles.

6.1 Introduction

This chapter presents a path planning algorithm that is not dependent on experience or prior knowledge [63, 64, 69] such that it would be robust against unknown new environments. Dynamic programming is adopted and the solution is globally optimal, compare with quasi optima in [65, 72, 75, 76, 119]. Instead of pixels or grids in [67, 68, 70], obstacle discs are used, and this would give an accurate environment description, in comparison with potential field in [72, 73, 76]. The disc obstacles are identified as convex clusters. This not only improves computation efficiency, but also provides a solution for path planning through non-convex obstacles, in comparison with non-cluster-based approach in [74].

6.2 Obstacle Filtering

Using generic image processing, physical obstacles can be identified as a cluster of obstacle points. We propose that a cluster of such obstacle points can be depicted by a obstacle disc, such as disc P_1, P_2, P_3 and P_4 in Fig 6.1. A large number of obstacle points can be denoted by a few discs, each of which can be represented by its center and a radius. Instead of computing point by point, algorithms can be executed disc by disc. Such that, efficiency is greatly improved and real-time response can be obtained.

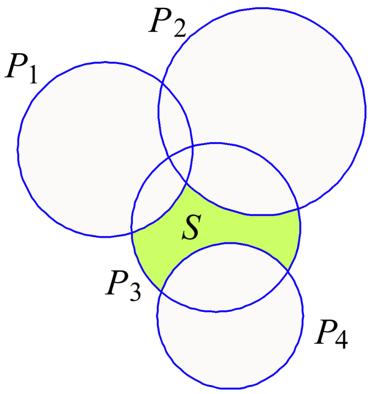


Figure 6.1: Four intersection obstacle discs.

However, transforming the points into disc could introduce repetitive data as well. Obstacle discs may have interactions. In the intersection area, points can be denoted by either of the interaction discs. Thus, it is possible that any point of a given disc can be denoted by a different disc. Then, we say that such given disc is invalid; otherwise, we say that it is valid. We can calculate the exclusive area of the intersection to determine whether a disc is valid. For example in Fig 6.1, P_3 has different intersections with P_1, P_2 and P_4. After calculating the exclusive area S of P_3, if $S = 0$, then P_3 is valid; otherwise, P_3 is invalid.

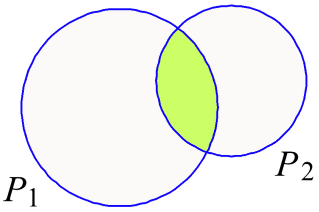


Figure 6.2: Two intersection obstacle discs.

The fundamental case is to find the exclusive area between two interacting discs in Fig 6.2. It is easy to calculate that the area of disc P_1 is S_1, and the area of disc P_2 is S_2. The common area of the intersection between P_1 and P_2 to be calculated is S_{12}. We can denote the exclusive area of P_1 as $S_1^2 = S_1 - S_{12}$ and the exclusive area of P_2 as $S_2^1 = S_2 - S_{12}$.

To start with, we find the exclusive area between two interacting discs in Fig 6.2. It is easy to find that the area of disc P_1 is S_1, and the area of disc P_2 is S_2. The common area of the intersection between P_1 and P_2 to be calculated is S_{12}. We denote the exclusive area of P_1 as $S_1^2 = S_1 - S_{12}$ and the exclusive area of P_2 as $S_2^1 = S_2 - S_{12}$.

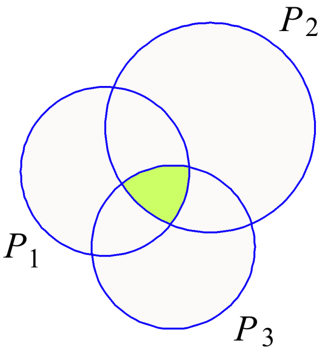


Figure 6.3: Three intersection obstacle discs.

Note that Fig 6.3 shows the case of three interacting discs. Similarly, it is easy to find that the area of disc P_1 is S_1. The common areas to be calculated are S_{12} between P_1 and P_2, S_{13} between P_1 and P_3, and S_{123} among P_1, P_2 and P_3. We denote the exclusive area of P_1 as $S_1^{23} = S_1 - S_{12} - S_{13} + S_{123}$.

Note that Fig 6.4 shows the case of four interacting discs. In this case, we consider the collection of discs P_1 and P_4 as a disc cluster $P_{1'}$. To find the exclusive area $S_3^{1'2}$ for P_3, we have $S_3^{1'2} = S_3 - S_{31'} - S_{32} + S_{31'2}$, and $S_{31'}$ and $S_{31'2}$ are to be calculated. As a matter of fact, we have $S_{31'} = S_{13} + S_{34} - S_{134}$ and $S_{31'2} = S_{312} + S_{342} - S_{1234}$.

It can be seen that the common areas S_{12} and S_{13} between two discs can be used to calculate the exclusive area S_1^{23} among three discs, the common areas S_{13}, S_{34}, S_{134}, S_{123}, S_{234} among two and three discs can be used to calculate the exclusive area $S_3^{1'2}$ among four discs. The problem that remains is to find the common area $S_{123\dots}$ between any given number of discs P_1, P_2, P_3, ….

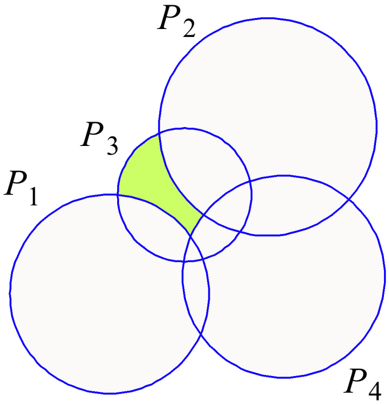


Figure 6.4: Four intersection obstacle discs.

Note that Fig 6.5 demonstrates how the common area among any given number of discs can be calculated. In fact, the common area S of the five discs is $S = S_{Q_1Q_2Q_3Q_4Q_5} + S_{\overset{\frown}{Q_1Q_2}} + S_{\overset{\frown}{Q_2Q_3}} + S_{\overset{\frown}{Q_3Q_4}} + S_{\overset{\frown}{Q_4Q_5}} + S_{\overset{\frown}{Q_5Q_1}}$, where $S_{Q_1Q_2Q_3Q_4Q_5}$ is the area of the polygon and $S_{\overset{\frown}{Q_1Q_2}}$ is the arched area enclosed by the line segment Q_1Q_2 and the curve $\overset{\frown}{Q_1Q_2}$.

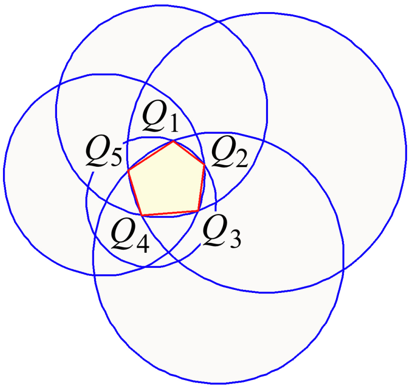


Figure 6.5: Common area of five intersection obstacle discs.

An Illustration

A group of fifteen obstacle discs is shown in Fig 6.6. It can be clearly seen that there are multiple invalid discs. We can apply the filtering algorithm to obtain a minimum subset of the obstacle collection. The result is demonstrated in Fig 6.7, showing that after the filtering, seven invalid disc obstacles are eliminated. The remaining eight disc obstacles are sufficient for further processing.

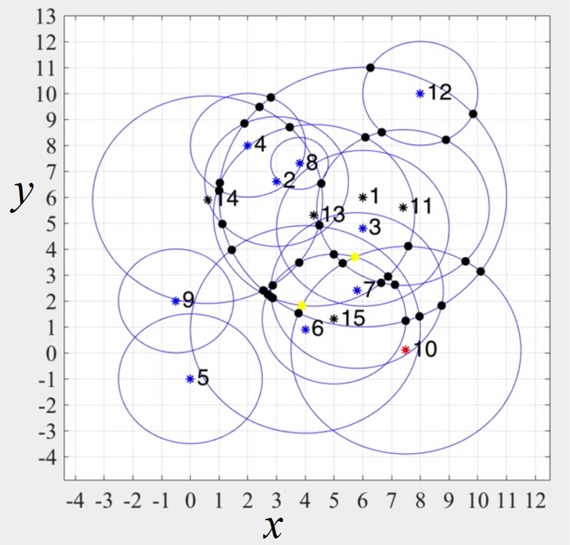


Figure 6.6: Fifteen intersecting obstacle discs, including invalid discs.

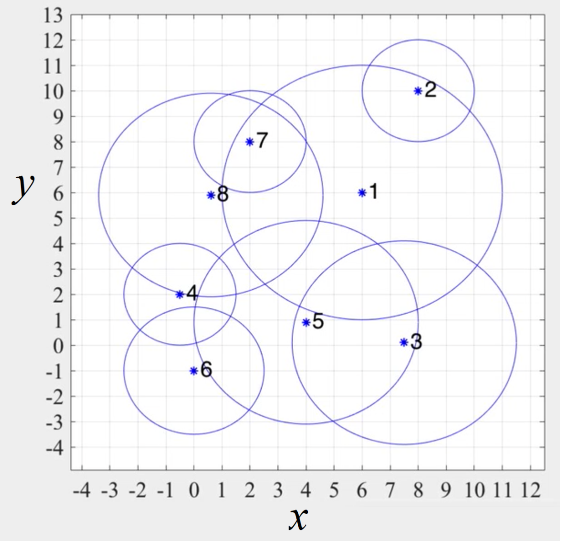


Figure 6.7: Eight intersecting obstacle discs after filtering.

6.3 Cluster Identification

In the previous section, we have designed a scheme to filter the invalid discs. The reduced obstacle discs help to reduce computation burden and improve real-time response. In this section, we design classifiers to identify appropriate obstacle disc clusters. These clusters are used in the following section in optimal path planning.

The fundamental case involves two obstacle discs, and the two need to be intersected to be considered as a cluster. The intersection gives us six points Q_1, Q_2, …, Q_6, which line up to be the contour and possible paths around the two discs. We propose that the disc conglomerate should be convex, and this means that the contour points should be convex. This always holds for two intersecting discs.

Note that Fig 6.9 shows the case with three discs. The contour for the two intersecting conglomerates are $C_1 = Q_1Q_3Q_4Q_5Q_7Q_{11}Q_1$ and $C_2 = Q_{12}Q_{10}Q_9Q_8Q_6Q_2Q_{12}$ following the correct orders. It can be seen that line segment Q_6Q_2 on C_2 is completely inside C_1 while Q_9Q_8 is not. We propose that for any line segments $Q_{1'}Q_{2'}$ in C_2 that is not completely inside C_1, if C_1 partially falls on both the left and the right sides of $Q_{1'}Q_{2'}$, then C_1 and C_2 cannot establish a convex cluster. This also means that the three discs can not establish a convex cluster.

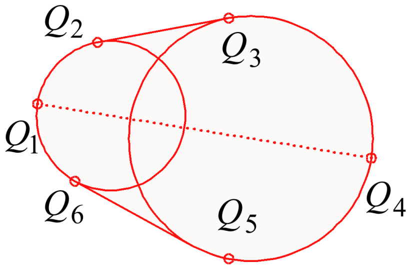


Figure 6.8: A convex cluster of two obstacles.

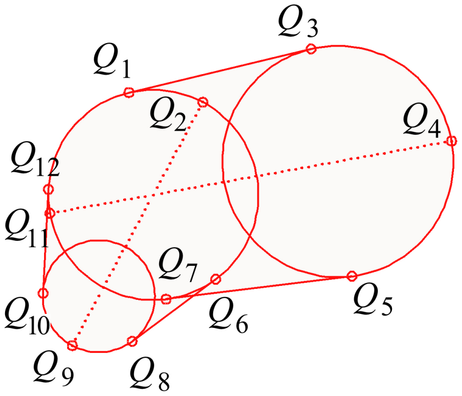


Figure 6.9: A non-convex cluster of three obstacles.

To derive all the possible convex clusters, we start with each and every obstacle disc. Given a disc P_1, we find all the two-disc clusters including P_1. Then, we derive all the three-disc clusters including the two discs already found. Continuing the process, we can find all the convex clusters containing P_1. After excluding all the repetitive clusters and retaining the largest clusters, we obtain all the possible clusters, which include all the obstacle discs.

6.4 Optimal Path Finding

In the previous section, we have classified the filtered obstacle discs into clusters. In this section, we derive the optimal path from a starting point to a finishing point subject to multiple obstacle clusters.

To derive the optimal path, we first find the path from a point to a obstacle cluster. In Fig 6.10, six paths can be identified, and they are Q_1Q_2, Q_1Q_3,

Q_1Q_4, Q_1Q_5, Q_1Q_6 and Q_1Q_7. Of these paths, only Q_1Q_2, Q_1Q_6, Q_1Q_3 and Q_1Q_7 are eligible. Moreover, Q_1Q_2 and Q_1Q_6 should be discarded as they are clipped by Q_1Q_3 and Q_1Q_7. Thus, the paths from Q_1 to the three-disc cluster are Q_1Q_3 and Q_1Q_7.

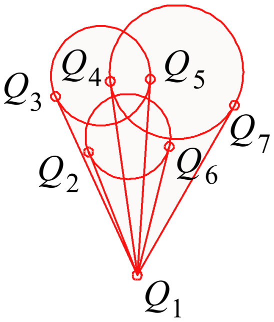


Figure 6.10: Possible paths from a point to a cluster.

Find the path from a obstacle cluster to another cluster is the next step. In Fig 6.11, the paths between the two clusters are Q_1Q_2, Q_3Q_4, Q_5Q_6 and Q_7Q_8. For a path to be eligible, it needs to neatly divide the contour points of the two clusters. For example, all the contour points of the two clusters fall on the same side of Q_1Q_2 and Q_7Q_8, and all the points of the two clusters fall on different sides of Q_3Q_4 and Q_5Q_6.

It is easy to find the paths between two points on the contour of an obstacle cluster. In Fig 6.12, the contour points of the three-disc cluster is $Q_3Q_4Q_5Q_6Q_7Q_8Q_9Q_{10}$ in a correct order. For two points Q_1 and Q_2 on the contour, there are two paths $Q_1Q_3Q_4Q_5Q_6Q_7Q_8Q_2$ and $Q_1Q_{10}Q_9$.

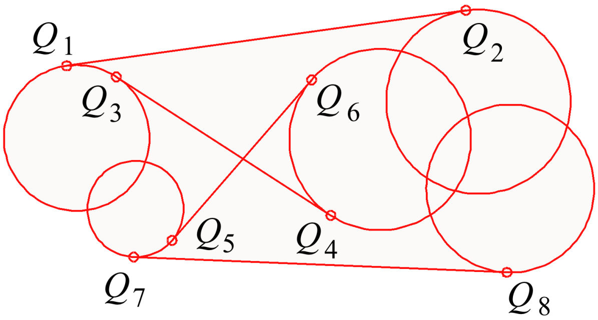


Figure 6.11: Possible paths between two clusters.

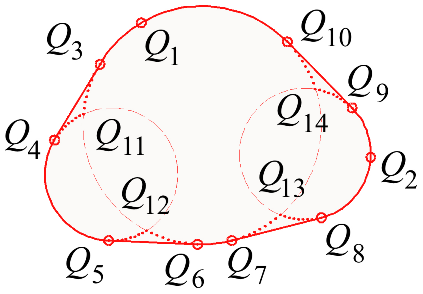


Figure 6.12: Possible paths within one cluster.

We have presented the scheme to derive paths between a point and a cluster, paths between clusters and paths within a cluster. Next, we derive the optimal path between two clusters C_i and C_j (point is regarded as the smallest cluster) through multiple obstacle clusters C_1, C_2, ... The following procedure is proposed.

1) Check whether the path C_iC_j is clear of obstacles. If it is, then the optimal path is C_iC_j; otherwise, go to 2).

2) Derive all the clear paths C_jO_{j1}, C_jO_{j2}, ... from the clusters C_1, C_2, ... to C_j. Then, derive all the clear paths C_iO_{i1}, C_iO_{i2}, ... from C_i to the clusters C_1, C_2, ... If there is no such clear path, then go to 3.1); otherwise, go to 3.2);

3.1) For a clear path C_jO_{j1}, go to 1) and derive the paths from the clusters C_2, C_3 ... to cluster O_{j1}.

3.2) Derive the clear paths between O_{j1} and O_{i1} on C_1, O_{j2} and O_{i2} on C_2, ... If there is no such clear path, then go to 3.1); otherwise the shortest path can be derived from current paths from C_i to C_j.

The example of an optimal path is given in Fig 6.13. It can be seen that there is no clear path from C_i to C_j and the path from C_1 to C_j is found. Again, there is no clear path from C_i to C_1, and the path from C_2 to C_1 is found. Still, there is no clear path from C_i to C_2, and the path from C_4 to C_2 is found. Finally, the path from C_i to C_4 is found. The optimal path thus runs along $C_iC_4C_2C_1C_j$.

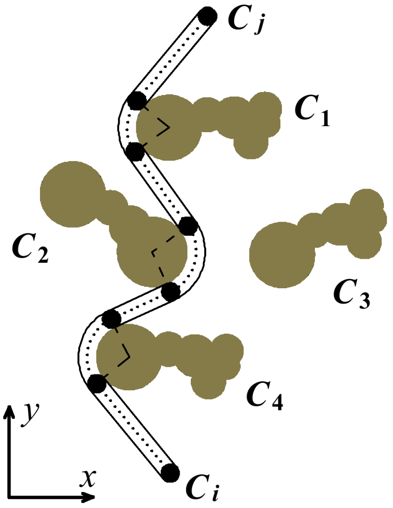


Figure 6.13: The optimal path between two points through four obstacle clusters.

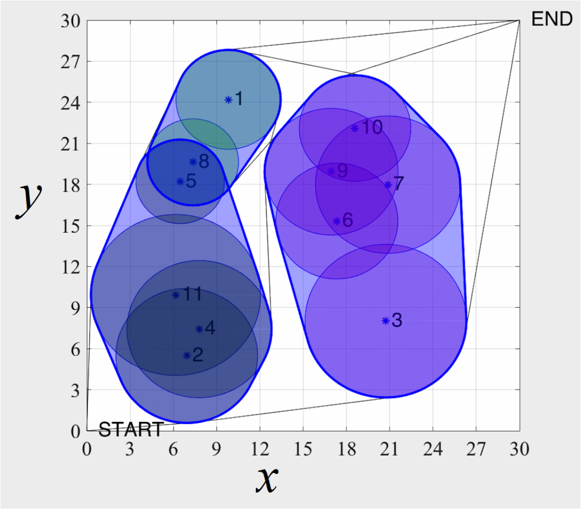


Figure 6.14: Possible paths among two points and three obstacle clusters.

One example of obstacle identifying and optimal path planning is given in Figs 6.14 and 6.15. It can be seen that under the identifying algorithm, discs 1, 8 and 5 are regarded as cluster 3, discs 2, 4, 5 and 11 are regarded as cluster 2, and discs 3, 6, 7, 9 and 10 are regarded as cluster 1. The optimal path is coloured in red in Fig 6.15 through the three obstacle clusters.

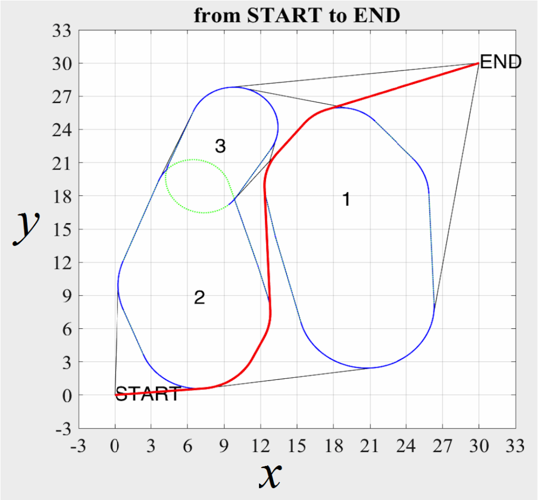


Figure 6.15: The red optimal path through three obstacle clusters.

6.5 Path Following

In the previous section, a path planning scheme is proposed and in this chapter, we design control laws to drive the robot along the path. In order to do so, we first establish the dynamic of the robot. We assume that the robot has little inertia and the motors are strong enough to directly manipulate the wheel speeds. The speeds of the wheels are instantly changed at each sampling.

Denote at instant k the speeds of the left and right wheels are $v_l(k)$ and $v_r(k)$; the speed of the robot center is $v_c(k)$ the radius of the disc robot is R; the sampling rate is τ; the orientation and position of the robot is $e_r(k)$ and $x(k)$. Thus, the dynamic equations of the robot are given below.

$$\begin{aligned}
v_c(k) &= \frac{v_l(k)+v_r(k)}{2}, \quad r_c(k) = R\frac{v_l(k)+v_r(k)}{v_l(k)-v_r(k)}, \\
\theta(k) &= \frac{v_c(k)\tau}{r_c(k)}, \\
e_r(k+1) &= \left[\begin{array}{cc} \cos(\theta) & \sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{array}\right] e_r(k), \\
x(k+1) &= x(k) + 2r_c(k)\frac{e_r(k)+e_r(k+1)}{|e_r(k)+e_r(k+1)|}\sin\frac{\theta}{2},
\end{aligned} \tag{6.1}$$

where $\theta(k)$ is the deviation angle and $r_c(k)$ is the arc radius.

We design the following control law below to manipulate the speeds of the left and right wheels. We assume that the optimal path is $x(k)Q_1(k)Q_2(k)\dots Q$, where Q is the END point.

$$
\begin{aligned}
e(k) &= \frac{Q_1(k)-x(k)}{|Q_1(k)-x(k)|},\\
\omega(k) &= [e_r(k-1),0]\times[e_r(k),0]\,,\\
\omega_d(k) &= \omega_0\mathrm{sign}(e_r(k)e(k)')\,[e_r(k),0]\times[e(k),0]\,,\\
\phi(k) &= [e(k),0]\times\left[e(k)\begin{bmatrix}0 & 1\\ -1 & 0\end{bmatrix},0\right],\\
\epsilon(k) &= \begin{cases}1, & \text{if} \quad |v_l(k)+v_r(k)|<v_{up};\\ 0, & \text{if} \quad |v_l(k)+v_r(k)|\geq v_{up},\end{cases}\\
v_l(k+1) &= v_l(k)-\sigma_l^1\mathrm{sign}(\phi(k)(\omega_d(k)-\omega(k))'\\
&\quad +\sigma_l^2\epsilon(k)\mathrm{sign}(e_r(k)e(k)'),\\
v_r(k+1) &= v_r(k)+\sigma_r^2\mathrm{sign}(\phi(k)(\omega_d(k)-\omega(k))')\\
&\quad +\sigma_r^2\epsilon(k)\mathrm{sign}(e_r(k)e(k)'), \qquad (6.2)
\end{aligned}
$$

where $e(k)$ is the unit direction vector from $x(k)$ to $Q_1(k)$, $\omega(k)$ is the rotational vector, $\omega_d(k)$ is the desired rotational vector, ω_0 is the desired rotational speed, $\phi(k)$ is the upright rotational vector, $\epsilon(k)$ is the threshold for speed control, and v_{up} is the speed upper bound. For the left and right wheels, σ_l^1 and σ_r^1 are the minimal shift change, and σ_l^2 and σ_r^2 are the minimal rotation change.

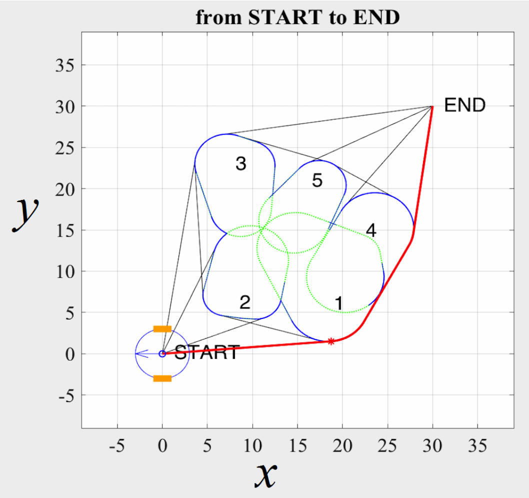


Figure 6.16: Robot at the START point follows the red optimal path.

The example of an optimal path is given. Note that we have taken the values for equations (6.1) and (6.2) with $R = 3\text{m}$, $\tau = 0.01\text{s}$, $\omega_0 = 0.1$, $v_{up} = 20\text{m/s}$, $\sigma_l^1 = 0.5$, $\sigma_r^1 = 0.5$, $\sigma_l^2 = 1$, and $\sigma_r^2 = 1$. The optimal path is shown in Figs 6.16, 6.17 and 6.18, and it can be seen that under the control law, the robot is capable of following the optimal path. The cures of the left and right wheel speeds are shown in Fig 6.19.

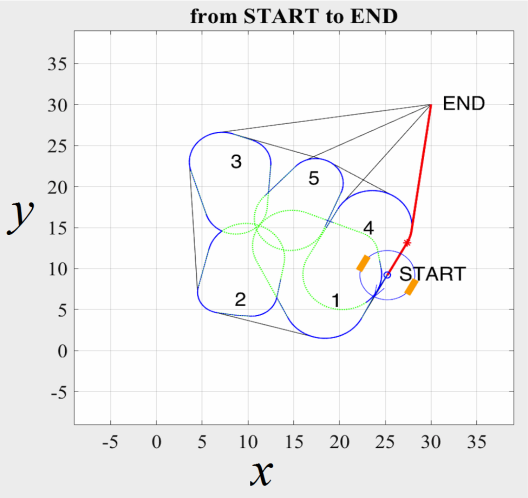


Figure 6.17: Robot on the way following the red optimal path.

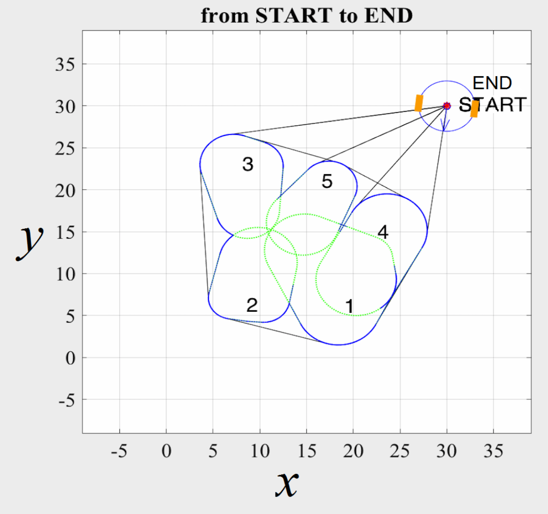


Figure 6.18: Robot eventually reaches the END point.

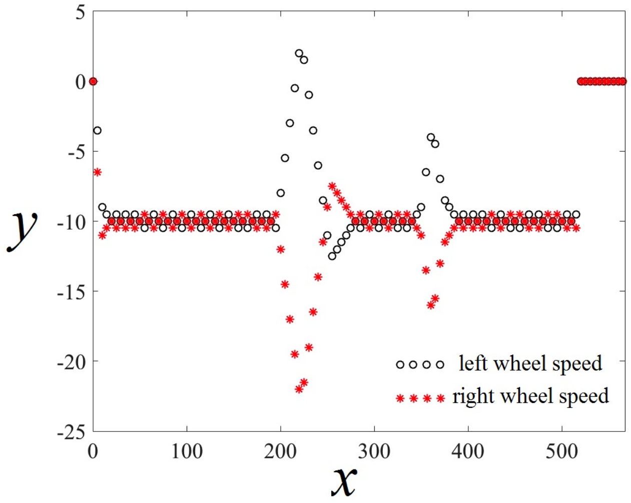


Figure 6.19: The curves of the left and right wheel speeds.

6.6 Summary

In this chapter, disc obstacles are used in path planning to represent obstacles points and improve real-time response. Obstacle filter is proposed to eliminate the invalid discs induced from obstacle intersections. Based on the overlapping disc obstacles, convex disc clusters are identified. Three different cases are studied to derive the optimal paths from a point to a disc cluster, between two disc clusters and within one disc cluster. Dynamic programming is used to derive the optimal path based on the convex disc clusters.

Chapter 7

Conclusion

Formation control of robotic swarm is considered in this thesis. The algorithms feature localised and decentralised structure and distributed computing. For first-order systems, Chapter 2 presents two coordination protocols for robotic formation with limited sensing and communication. It has been demonstrated that one protocol is efficient in driving the robots to a global one-to-one mapping and the other achieves improvements on shortening overall wandering paths. The schemes are proved to be suitable for second-order systems. For future work, scalability of the algorithms can be investigated based on the three types of links and the three types of the topologies. Performance improvements can be investigated when the robots are capable of complicated communication. Formation control of heterogeneous robots is also a potential focus of research.

To deal with the lack of communication, Chapter 3 proposes a matching based coordination algorithm, the mapping of which enables a novel controller to drive the robots to a desired formation. The objective formation is designed based on displacement rules such that the swarm can incorporate a varying number of robots. Under the controller, the swarm is able to reach a faulty formation with a faulty mapping. Then, the protocol identifies the faults and make a remapping. Under the scalable objective formation, all the formation and mapping faults are eliminated and the desired formation is achieved. For future work, similar problem setup can be studied for robots with anisotropic sensing abilities. This opens for applications in confined or hostile environments. Active connectivity maintaining can be investigated for incoming and outgoing robots in congested operation areas. Advanced stability analysis tools can be developed to push the system performance to its borders.

For robotic swarm of fixed size, Chapter 4 investigates a two-level optimisation strategy for the individual robots to identify their position in the formation and make appropriate mapping decisions. The mapping loops are gradually cancelled after introducing probability triggered control law. Minimum local topology is established for the controller, and as a result, the topology has gained improving robustness. For future work, different roles can be assigned for different robots, and position allocation is needed for those that share the same roles. Memories can be introduced to substitute the probability component in the algorithm and provide guaranteed performance. To adapt to a larger variety of tasks, a library of desired formations can be designed and an algorithm can be proposed to select the most suitable desired one. Each desired formation is most suitable for one task and this broadens the applications of the proposed algorithms.

Chapter 5 attempts to implement a practical formation scheme on multi-robot platform composed of two-wheel Arduino robots. A controller is designed to combat communication delay, occasional motor failure and static friction. Instead of relying on speed sensors, the prediction on trajectory produced is based on positions and orientations. The followers aims to corner the leader in an adaptive fashion. Computation is distributed and collision avoidance is locally achieved. For future work, the proposed algorithms can be modified and applied to three dimensional environments. Some schedule control have been used and the algorithms can be used in applications, which requires coordinated and synchronised operations. Cooperation among the robots are investigated and this can be expanded to collective path planning in a confined environment.

To operate in a complex environment, Chapter 6 studies path planning for globally optimal solution. Obstacle discs are used to describe the congested area. The intersections among these discs are used to form convex clusters. In a similar way, a non-convex cluster can also be transformed into convex ones. The convex clusters provide paths from point to cluster, between clusters and within cluster. Dynamic programming then derives the optimal path from the start point to the end point through multiple obstacle discs. For future work, the two dimensional path planning can be expanded to the three dimensional space. Mobile obstacles can be dealt with separately to ensure minimal additional update to the existing clusters, which aims to improve the real-time responses. The online path planning can also be used to provide a comprehensive environmental evaluation on all possible paths, and the path library can be used in areas of vulnerability assessment, fault detection and multimodal transportation.

Bibliography

[1] Balch, T., & Arkin, R. C. (1998). Behavior-based formation control for multirobot teams. IEEE Transactions on Robotics and Automation, 14(6), 926-939.

[2] Carrascosa, C., Bajo, J., Julin, V., Corchado, J. M., & Botti, V. (2008). Hybrid multi-agent architecture as a real-time problem-solving model. Expert Systems with Applications, 34(1), 2-17.

[3] Calafiore, G. C., Carlone, L., & Wei, M. (2012). A distributed technique for localization of agent formations from relative range measurements. IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans, 42(5), 1065-1076.

[4] Das, A. K., Fierro, R., Kumar, V., Ostrowski, J. P., Spletzer, J., & Taylor, C. J. (2002). A vision-based formation control framework. IEEE Transactions on Robotics and Automation, 18(5), 813-825.

[5] Du, H., Li, S., & Shi, P. (2012). Robust consensus algorithm for second-order multi-agent systems with external disturbances. International Journal of Control, 85(12), 1913-1928.

[6] Desai, J. P., Ostrowski, J. P., & Kumar, V. (2001). Modeling and control of formations of nonholonomic mobile robots. IEEE Transactions on Robotics and Automation, 17(6), 905-908.

[7] Do, K. D. (2007). Bounded controllers for formation stabilization of mobile agents with limited sensing ranges. IEEE Transactions on Automatic Control, 52(3), 569-576.

[8] Do, K. D. (2012). Formation control of multiple elliptical agents with limited sensing ranges. Automatica, 48(7), 1330-1338.

[9] Fatima, S. S., Wooldridge, M., & Jennings, N. R. (2002, July). Multi-issue negotiation under time constraints. In Proceedings of the First

International Joint Conference on Autonomous Agents and Multiagent Systems, Part 1 (pp. 143-150). ACM.

[10] Guo, G., & Yue, W. (2012). Autonomous platoon control allowing range-limited sensors. IEEE Transactions on Vehicular Technology, 61(7), 2901-2912.

[11] Jin, Y., Guo, H., & Meng, Y. (2012). A hierarchical gene regulatory network for adaptive multirobot pattern formation. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 42(3), 805-816.

[12] Kwon, J. W., & Chwa, D. (2012). Hierarchical formation control based on a vector field method for wheeled mobile robots. IEEE Transactions on Robotics, 28(6), 1335-1345.

[13] Kwon, J. W., Kim, J. H., & Seo, J. (2015). Multiple leader candidate and competitive position allocation for robust formation against member robot faults. Sensors, 15(5), 10771-10790.

[14] Kan, Z., Navaravong, L., Shea, J. M., Pasiliao, E. L., & Dixon, W. E. (2015). Graph matching-based formation reconfiguration of networked agents with connectivity maintenance. IEEE Transactions on Control of Network Systems, 2(1), 24-35.

[15] Li, W., & Spong, M. W. (2014). Analysis of flocking of cooperative multiple inertial agents via a geometric decomposition technique. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 44(12), 1611-1623.

[16] Li, Z., Ren, W., Liu, X., & Fu, M. (2013). Distributed containment control of multiagent systems with general linear dynamics in the presence of multiple leaders. International Journal of Robust and Nonlinear Control, 23(5), 534-547.

[17] Moreau, L. (2005). Stability of multiagent systems with time-dependent communication links. IEEE Transactions on Automatic Control, 50(2), 169-182.

[18] Mei, J., Ren, W., & Ma, G. (2013). Distributed coordination for second-order multi-agent systems with nonlinear dynamics using only relative position measurements. Automatica, 49(5), 1419-1427.

[19] Mao, Y., Dou, L., Fang, H., & Chen, J. (2015). Bounded connectivity-preserving leader-follower flocking algorithms without acceleration measurements. Asian Journal of Control, 17(1), 304-314.

[20] Ni, W., Wang, X., & Xiong, C. (2013). Consensus controllability, observability and robust design for leader-following linear multi-agent systems. Automatica, 49(7), 2199-2205.

[21] Olfati-Saber, R., & Murray, R. M. (2004). Consensus problems in networks of agents with switching topology and time-delays. IEEE Transactions on Automatic Control, 49(9), 1520-1533.

[22] Park, B. S., Park, J. B., & Choi, Y. H. (2011). Adaptive formation control of electrically driven nonholonomic mobile robots with limited information. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 41(4), 1061-1075.

[23] Sierra, C., Faratin, P., & Jennings, N. R. (1999). A service-oriented negotiation model between autonomous agents. In Collaboration Between Human and Artificial Societies (pp. 201-219). Springer Berlin Heidelberg.

[24] Shi, P., & Shen, Q. (2015). Cooperative control of multi-agent systems with unknown state-dependent controlling effects. IEEE Transactions on Automation Science and Engineering, 12(3), 827-834.

[25] Shen, Q., Jiang, B., Shi, P., & Zhao, J. (2014). Cooperative adaptive fuzzy tracking control for networked unknown nonlinear multiagent systems with time-varying actuator faults. IEEE Transactions on Fuzzy Systems, 22(3), 494-504.

[26] Wen, G., Duan, Z., Ren, W., & Chen, G. (2014). Distributed consensus of multiagent systems with general linear node dynamics and intermittent communications. International Journal of Robust and Nonlinear Control, 24(16), 2438-2457.

[27] Wang, X., Li, S., & Shi, P. (2014). Distributed finite-time containment control for double-integrator multiagent systems. IEEE Transactions on Cybernetics, 44(9), 1518-1528.

[28] Wang, X., Ni, W., & Wang, X. (2012). Leader-following formation of switching multirobot systems via internal model. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 42(3), 817-826.

[29] Yu, W., Ren, W., Zheng, W. X., Chen, G., & L, J. (2013). Distributed control gains design for consensus in multi-agent systems with second-order nonlinear dynamics. Automatica, 49(7), 2107-2115.

[30] Zakhar'eva, A., Matveev, A. S., Hoy, M. C., & Savkin, A. V. (2015). Distributed control of multiple non-holonomic robots with sector vision and range-only measurements for target capturing with collision avoidance. Robotica, 33(02), 385-412.

[31] Cao, M., Yu, C., & Anderson, B. D. (2011). Formation control using range-only measurements. Automatica, 47(4), 776-781.

[32] Cao, Y., & Ren, W. (2010). Multivehicle coordination for doubleintegrator dynamics under fixed undirected/directed interaction in a sampled-data setting. International Journal of Robust and Nonlinear Control, 20(9), 987-1000.

[33] Cao, Y., & Ren, W. (2014). Finite-time consensus for multi-agent networks with unknown inherent nonlinear dynamics. Automatica, 50(10), 2648-2656.

[34] Dimarogonas, D. V., & Johansson, K. H. (2010). Stability analysis for multi-agent systems using the incidence matrix: Quantized communication and formation control. Automatica, 46(4), 695-700.

[35] Dimarogonas, D. V., & Johansson, K. H. (2009, June). Further results on the stability of distance-based multi-robot formations. In American Control Conference, 2009. ACC'09. (pp. 2972-2977). IEEE.

[36] Hao, H., Barooah, P., & Mehta, P. G. (2011). Stability margin scaling laws for distributed formation control as a function of network structure. IEEE Transactions on Automatic Control, 56(4), 923-929.

[37] Liu, T., & Jiang, Z. P. (2013). Distributed formation control of nonholonomic mobile robots without global position measurements. Automatica, 49(2), 592-600.

[38] Li, Z., Ren, W., Liu, X., & Fu, M. (2013). Consensus of multi-agent systems with general linear and Lipschitz nonlinear dynamics using distributed adaptive protocols. IEEE Transactions on Automatic Control, 58(7), 1786-1791.

[39] Lin, Z., Wang, L., Han, Z., & Fu, M. (2014). Distributed formation control of multi-agent systems using complex Laplacian. IEEE Transactions on Automatic Control, 59(7), 1765-1777.

[40] Falconi, R., Sabattini, L., Secchi, C., Fantuzzi, C., & Melchiorri, C. (2015). Edge-weighted consensus-based formation control strategy with collision avoidance. Robotica, 33(02), 332-347.

[41] Hao, H., & Barooah, P. (2013). Stability and robustness of large platoons of vehicles with doubleintegrator models and nearest neighbor interaction. International Journal of Robust and Nonlinear Control, 23(18), 2097-2122.

[42] Huang, H., Yu, C., & Wu, Q. (2013). Autonomous scale control of multiagent formations with only shape constraints. International Journal of Robust and Nonlinear Control, 23(7), 765-791.

[43] Laventall, K., & Corts, J. (2009). Coverage control by multi-robot networks with limited-range anisotropic sensory. International Journal of Control, 82(6), 1113-1121.

[44] Meng, Y., Guo, H., & Jin, Y. (2013). A morphogenetic approach to flexible and robust shape formation for swarm robotic systems. Robotics and Autonomous Systems, 61(1), 25-38.

[45] Sabattini, L., Secchi, C., & Fantuzzi, C. (2011). Arbitrarily shaped formations of mobile robots: Artificial potential fields and coordinate transformation. Autonomous Robots, 30(4), 385-397.

[46] Viegas, D., Batista, P., Oliveira, P., & Silvestre, C. (2016). Decentralized state observers for rangebased position and velocity estimation in acyclic formations with fixed topologies. International Journal of Robust and Nonlinear Control, 26(5), 963-994.

[47] Yu, H., Shi, P., & Lim, C. C. (2016). Robot formation control in stealth mode with scalable team size. International Journal of Control, 89(11), 2155-2168.

[48] Yu, H., Shi, P., & Lim, C. C. (2017). Scalable formation control in stealth with limited sensing range. International Journal of Robust and Nonlinear Control, 27(3), 410-433.

[49] Yue, W., Guo, G., Wang, L., & Wang, W. (2015). Nonlinear platoon control of Arduino cars with range-limited sensors. International Journal of Control, 88(5), 1037-1050.

[50] Abbaspour, A., Alipour, K., Jafari, H. Z., & Moosavian, S. A. A. (2015). Optimal formation and control of cooperative wheeled mobile robots. Comptes Rendus Mcanique, 343(5), 307-321.

[51] Aranda, M., Lpez-Nicols, G., Sags, C., & Mezouar, Y. (2015). Formation control of mobile robots using multiple aerial cameras. IEEE Transactions on Robotics, 31(4), 1064-1071.

[52] Cepeda-Gomez, R., & Perico, L. F. (2015). Formation control of nonholonomic vehicles under time delayed communications. IEEE Transactions on Automation Science and Engineering, 12(3), 819-826.

[53] Cifuentes, S., Girn-Sierra, J. M., & Jimnez, J. (2015). Virtual fields and behaviour blending for the coordinated navigation of robot teams: Some experimental results. Expert Systems with Applications, 42(10), 4778-4796.

[54] Chen, X., & Jia, Y. (2015). Adaptive leader-follower formation control of non-holonomic mobile robots using active vision. IET Control Theory & Applications, 9(8), 1302-1311.

[55] Degener, B., Kempkes, B., Kling, P., & Heide, F. M. A. D. (2015). Linear and competitive strategies for continuous robot formation problems. ACM Transactions on Parallel Computing, 2(1), 2.

[56] Ferreira, B. M., Matos, A. C., Cruz, N. A., & Moreira, A. P. (2016). Coordination of marine robots under tracking errors and communication constraints. IEEE Journal of Oceanic Engineering, 41(1), 27-39.

[57] Loria, A., Dasdemir, J., & Jarquin, N. A. (2016). Leaderfollower formation and tracking control of mobile robots along straight paths. IEEE Transactions on Control Systems Technology, 24(2), 727-732.

[58] Lin, C. W., Khong, M. H., & Liu, Y. C. (2015). Experiments on human-in-the-loop coordination for multirobot system with task abstraction. IEEE Transactions on Automation Science and Engineering, 12(3), 981-989.

[59] Poonawala, H. A., Satici, A. C., Eckert, H., & Spong, M. W. (2015). Collision-free formation control with decentralized connectivity preservation for nonholonomic-wheeled mobile robots. IEEE Transactions on Control of Network Systems, 2(2), 122-130.

[60] Park, M. C., & Ahn, H. S. (2015). Stabilisation of directed cycle formations and application to two-wheeled mobile robots. IET Control Theory & Applications, 9(9), 1338-1346.

[61] Reyes, L. A. V., & Tanner, H. G. (2015). Flocking, formation control, and path following for a group of mobile robots. IEEE Transactions on Control Systems Technology, 23(4), 1268-1282.

[62] Wang, Y., Cheng, L., Hou, Z. G., Yu, J., & Tan, M. (2016). Optimal formation of multirobot systems based on a recurrent neural network. IEEE Transactions on Neural Networks and Learning Systems, 27(2), 322-333.

[63] Berenson, D., Abbeel, P., & Goldberg, K. (2012, May). A robot path planning framework that learns from experience. In Robotics and Automation (ICRA), 2012 IEEE International Conference on (pp. 3671-3678). IEEE.

[64] Lim, Z. W., Hsu, D., & Lee, W. S. (2016). Adaptive informative path planning in metric spaces. The International Journal of Robotics Research, 35(5), 585-598.

[65] Roberge, V., Tarbouchi, M., & Labont, G. (2013). Comparison of parallel genetic algorithm and particle swarm optimization for real-time UAV path planning. IEEE Transactions on Industrial Informatics, 9(1), 132-141.

[66] Wang, X., Shi, Y., Ding, D., & Gu, X. (2016). Double global optimum genetic algorithm particle swarm optimization-based welding robot path planning. Engineering Optimization, 48(2), 299-316.

[67] Tuncer, A., & Yildirim, M. (2012). Dynamic path planning of mobile robots with improved genetic algorithm. Computers & Electrical Engineering, 38(6), 1564-1572.

[68] Gammell, J. D., Srinivasa, S. S., & Barfoot, T. D. (2014, September). Informed RRT*: Optimal sampling-based path planning focused via direct sampling of an admissible ellipsoidal heuristic. In Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on (pp. 2997-3004). IEEE.

[69] Nieuwenhuisen, M., & Behnke, S. (2016). Layered mission and path planning for MAV navigation with partial environment knowledge. In Intelligent Autonomous Systems 13 (pp. 307-319). Springer International Publishing.

[70] Cui, R., Li, Y., & Yan, W. (2016). Mutual information-based multi-AUV path planning for scalar field sampling using multidimensional

RRT. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 46(7), 993-1004.

[71] Raja, P., & Pugazhenthi, S. (2012). Optimal path planning of mobile robots: A review. International Journal of Physical Sciences, 7(9), 1314-1320.

[72] Ji, J., Khajepour, A., Melek, W., & Huang, Y. (2016). Path planning and tracking for vehicle collision avoidance based on model predictive control with multi-constraints. IEEE Transactions on Vehicular Technology.

[73] Wang, M., Shan, H., Lu, R., Zhang, R., Shen, X., & Bai, F. (2015). Real-time path planning based on hybrid-VANET-enhanced transportation system. IEEE Transactions on Vehicular Technology, 64(5), 1664-1678.

[74] Zhang, Y., Gong, D. W., & Zhang, J. H. (2013). Robot path planning in uncertain environment using multi-objective particle swarm optimization. Neurocomputing, 103, 172-185.

[75] Liu, M. (2016). Robotic online path planning on point cloud. IEEE Transactions on Cybernetics, 46(5), 1217-1228.

[76] Chen, Y. B., Luo, G. C., Mei, Y. S., Yu, J. Q., & Su, X. L. (2016). UAV path planning using artificial potential field method updated by optimal control theory. International Journal of Systems Science, 47(6), 1407-1420.

[77] Olfati-Saber, R., & Murray, R. M. (2002, December). Graph rigidity and distributed formation stabilization of multi-vehicle systems. In Decision and Control, 2002, Proceedings of the 41st IEEE Conference on (Vol. 3, pp. 2965-2971). IEEE.

[78] Qin, L., He, X., & Zhou, D. H. (2016). Fault-tolerant cooperative output regulation for multi-vehicle systems with sensor faults. International Journal of Control, 1-22.

[79] Bansal, S., Chen, M., Fisac, J. F., & Tomlin, C. J. (2016). Safe sequential path planning of multi-vehicle systems under disturbances and imperfect information. arXiv preprint arXiv:1603.05208.

[80] Wu, Y., Li, S. E., Zheng, Y., & Hedrick, J. K. (2016, December). Distributed sliding mode control for multi-vehicle systems with positive definite topologies. In Decision and Control (CDC), 2016 IEEE 55th Conference on (pp. 5213-5219). IEEE.

[81] Han, T., Lin, Z., & Zheng, R. (2016, August). A survey on concurrent network localization for autonomous multi-vehicle systems. In Guidance, Navigation and Control Conference (CGNCC), 2016 IEEE Chinese (pp. 1403-1408). IEEE.

[82] Chen, M., Shih, J. C., & Tomlin, C. J. (2016, December). Multi-vehicle collision avoidance via Hamilton-Jacobi reachability and mixed integer programming. In Decision and Control (CDC), 2016 IEEE 55th Conference on (pp. 1695-1700). IEEE.

[83] Harris, C. A., Phillips, A. B., Dopico-Gonzalez, C., & Brito, M. P. (2016, November). Risk and reliability modelling for multi-vehicle marine domains. In Autonomous Underwater Vehicles (AUV), 2016 IEEE/OES (pp. 286-293). IEEE.

[84] Qu, Y., Bekta, T., & Bennell, J. (2016). Sustainability SI: Multimode multicommodity network design model for intermodal freight transportation with transfer and emission costs. Networks and Spatial Economics, 16(1), 303-329.

[85] Baykasolu, A., & Subulan, K. (2016). A multi-objective sustainable load planning model for intermodal transportation networks with a real-life application. Transportation Research Part E: Logistics and Transportation Review, 95, 207-247.

[86] Ruan, J. H., Wang, X. P., Chan, F. T. S., & Shi, Y. (2016). Optimizing the intermodal transportation of emergency medical supplies using balanced fuzzy clustering. International Journal of Production Research, 54(14), 4368-4386.

[87] Shen, J., Cheng, C., Cheng, X., & Lund, J. R. (2016). Coordinated operations of large-scale UHVDC hydropower and conventional hydro energies about regional power grid. Energy, 95, 433-446.

[88] Zhang, B., Deng, W., Wang, T., Chen, J., Teng, Y., & Wu, X. (2016, October). Electromagnetic transient modeling and simulation of large-scale HVDC power grid with all primary devices. In Power and Renewable Energy (ICPRE), IEEE International Conference on Power (pp. 43-47). IEEE.

[89] Xu, C., & Abur, A. (2017). A fast and robust linear state estimator for very large scale interconnected power grids. IEEE Transactions on Smart Grid, DOI, 10.1109/TSG.2017.2676348.

[90] Ahlborn, M., Ahrens, J., & Schweickert, R. (2016). Large-scale transition of economic systems-do CEECs converge toward western prototypes. Comparative Economic Studies, 58(3), 430-454.

[91] Zhang, Y., Li, B., & Hong, J. (2016, April). Understanding user economic behavior in the city using large-scale geotagged and crowdsourced data. In Proceedings of the 25th International Conference on World Wide Web (pp. 205-214). International World Wide Web Conferences Steering Committee.

[92] Villavicencio, C., Schiaffino, S., Diaz-Pace, J. A., & Monteserin, A. (2016). PUMAS-GR: A negotiation-based group recommendation system for movies. In Advances in Practical Applications of Scalable Multi-agent Systems. The PAAMS Collection (pp. 294-298). Springer International Publishing.

[93] Su, S., Hong, J. P., Shi, J., & Park, H. S. (2016). Social behavior prediction from first person videos. arXiv preprint arXiv:1611.09464.

[94] Dickes, A. C., Sengupta, P., Farris, A. V., & Basu, S. (2016). Development of mechanistic reasoning and multilevel explanations of ecology in third grade using agent-based models. Science Education, 100(4), 734-776.

[95] Li, Z., Zeng, L., Zhong, H., & Wu, J. (2016, June). Research hotspots and trends in swarm intelligence: From 2000 to 2015. In International Conference in Swarm Intelligence (pp. 24-35). Springer International Publishing.

[96] Azar, A. T., & Vaidyanathan, S. (Eds.). (2016). Advances in chaos theory and intelligent control (Vol. 337). Springer.

[97] Hassanien, A. E., & Emary, E. (2016). Swarm intelligence: Principles, advances, and applications. CRC Press.

[98] Rini, D. P., Shamsuddin, S. M., & Yuhaniz, S. S. (2011). Particle swarm optimization: Technique, system and challenges. International Journal of Computer Applications, 14(1), 19-26.

[99] Mukhopadhyay, S., & Banerjee, S. (2012). Global optimization of an optical chaotic system by chaotic multi swarm particle swarm optimization. Expert Systems with Applications, 39(1), 917-924.

[100] Wen, G., Zhao, Y., Duan, Z., Yu, W., & Chen, G. (2016). Containment of higher-order multi-leader multi-agent systems: A dynamic output approach. IEEE Transactions on Automatic Control, 61(4), 1135-1140.

[101] Rahman, M. S., & Oo, A. M. T. (2017). Distributed multi-agent based coordinated power management and control strategy for microgrids with distributed energy resources. Energy Conversion and Management, 139, 20-32.

[102] Zuo, Z., & Tie, L. (2016). Distributed robust finite-time nonlinear consensus protocols for multi-agent systems. International Journal of Systems Science, 47(6), 1366-1375.

[103] Yu, W., Wen, G., Chen, G., & Cao, J. (2016). Delay-induced consensus and quasi-consensus in multi-agent systems. Distributed Cooperative Control of Multi-Agent Systems, 214-228.

[104] Yu, W., Li, Y., Wen, G., Yu, X., & Cao, J. (2017). Observer design for tracking consensus in second-order multi-agent systems: Fractional order less than two. IEEE Transactions on Automatic Control, 62(2), 894-900.

[105] He, W., Zhang, B., Han, Q. L., Qian, F., Kurths, J., & Cao, J. (2017). Leader-following consensus of nonlinear multiagent systems with stochastic sampling. IEEE Transactions on Cybernetics, 47(2), 327-338.

[106] Wang, W., Wen, C., & Huang, J. (2017). Distributed adaptive asymptotically consensus tracking control of nonlinear multi-agent systems with unknown parameters and uncertain disturbances. Automatica, 77, 133-142.

[107] Li, H., Liao, X., Huang, T., & Zhu, W. (2015). Event-triggering sampling based leader-following consensus in second-order multi-agent systems. IEEE Transactions on Automatic Control, 60(7), 1998-2003.

[108] Williams, R. K., & Sukhatme, G. S. (2013). Constrained interaction and coordination in proximity-limited multiagent systems. IEEE Transactions on Robotics, 29(4), 930-944.

[109] Song, L., & Wang, Y. (2014, August). Multiple target counting and tracking using binary proximity sensors: Bounds, coloring, and filter. In Proceedings of the 15th ACM International Symposium on Mobile ad Hoc Networking and Computing (pp. 397-406). ACM.

[110] Chand, P., & Carnegie, D. A. (2013). Mapping and exploration in a hierarchical heterogeneous multi-robot system using limited capability robots. Robotics and Autonomous Systems, 61(6), 565-579.

[111] Duchon, F., Babinec, A., Kajan, M., Beno, P., Florek, M., Fico, T., & Jurisica, L. (2014). Path planning with modified A star algorithm for a mobile robot. Procedia Engineering, 96, 59-69.

[112] Cheng, L., Liu, C., & Yan, B. (2014, July). Improved hierarchical A-star algorithm for optimal parking path planning of the large parking lot. In Information and Automation (ICIA), 2014 IEEE International Conference on (pp. 695-698). IEEE.

[113] Tseng, F. H., Liang, T. T., Lee, C. H., Der Chou, L., & Chao, H. C. (2014, August). A star search algorithm for civil UAV path planning with 3G communication. In Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP), 2014 Tenth International Conference on (pp. 942-945). IEEE.

[114] Salehinejad, H., Nezamabadi-Pour, H., Saryazdi, S., & Farrahi-Moghaddam, F. (2015). Combined A*-ants algorithm: A new multi-parameter vehicle navigation scheme. arXiv preprint arXiv:1504.07329.

[115] Dogar, M. R., Koval, M. C., Tallavajhula, A., & Srinivasa, S. S. (2014). Object search by manipulation. Autonomous Robots, 36(1-2), 153-167.

[116] Duan, H., & Qiao, P. (2014). Pigeon-inspired optimization: A new swarm intelligence optimizer for air robot path planning. International Journal of Intelligent Computing and Cybernetics, 7(1), 24-37.

[117] Muldoon, S. E., Luo, C., Shen, F., & Mo, H. (2014, December). Naturally inspired optimization algorithms as applied to mobile robotic path planning. In Swarm Intelligence (SIS), 2014 IEEE Symposium on (pp. 1-6). IEEE.

[118] Mo, H., & Xu, L. (2015). Research of biogeography particle swarm optimization for robot path planning. Neurocomputing, 148, 91-99.

[119] Wang, X., Shi, Y., Ding, D., & Gu, X. (2016). Double global optimum genetic algorithm particle-swarm optimization-based welding robot path planning. Engineering Optimization, 48(2), 299-316.

[120] Ni, J., Wang, K., Huang, H., Wu, L., & Luo, C. (2016, August). Robot path planning based on an improved genetic algorithm with variable length chromosome. In Natural Computation, Fuzzy Systems and Knowledge Discovery (ICNC-FSKD), 2016 12th International Conference on (pp. 145-149). IEEE.

[121] Kunzli, M., Meier, P., & Dornberger, R. (2016, September). A memory search algorithm for path finding problems compared with a genetic algorithm. In Computational and Business Intelligence (ISCBI), 2016 4th International Symposium on (pp. 66-73). IEEE.

[122] Lee, J., & Kim, D. W. (2016). An effective initialization method for genetic algorithm-based robot path planning using a directed acyclic graph. Information Sciences, 332, 1-18.

[123] Matveev, A. S., Semakova, A. A., & Savkin, A. V. (2016). Range-only based circumnavigation of a group of moving targets by a non-holonomic mobile robot. Automatica, 65, 76-89.

[124] Zhang, H., Zhao, G., & Xu, G. (2016, August). Three-dimensional optimal formation reconfiguration for multi-UAVs based on Chebyshev pseudospectral method. In Guidance, Navigation and Control Conference (CGNCC), 2016 IEEE Chinese (pp. 1253-1258). IEEE.

[125] Kamel, M. A., Yu, X., & Zhang, Y. (2016, June). Real-time optimal formation reconfiguration of multiple wheeled mobile robots based on particle swarm optimization. In Intelligent Control and Automation (WCICA), 2016 12th World Congress on (pp. 703-708). IEEE.

Statement of Authorship

Title of Paper	Distributed formation control with limited sensing ranges and scalability		
Publication Status	■ Published □ Submitted for Publication	□ Accepted for Publication □ Unpublished and Unsubmitted w ork w ritten in manuscript style	
Publication Details	Yu, H., Shi, P., & Lim, C. C.. Distributed formation control with limited sensing ranges and scalability. In Control Conference (AUCC), 2015 5th Australian (pp. 318-321). IEEE. 5-6 November 2015. Gold Coast, QLD, Australia.		

Principal Author

Name of Principal Author (Candidate)	Hongjun Yu		
Contribution to the Paper	Conducted theoretical derivation, performed practical analysis on all numerical examples, interpreted data, wrote manuscript and acted as corresponding author.		
Overall percentage (%)	90%		
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.		
Signature		Date	19/07/2017

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

i. the candidate's stated contribution to the publication is accurate (as detailed above);

ii. permission is granted for the candidate in include the publication in the thesis; and

iii. the sum of all co-author contributions is equal to 100% less the candidate's stated contribution.

Name of Co-Author	Peng Shi		
Contribution to the Paper	Supervised development of work, helped in data interpretation and manuscript evaluation.		
Signature		Date	19/07/2017

Name of Co-Author	Cheng-Chew Lim		
Contribution to the Paper	Supervised development of work, helped in data interpretation and manuscript evaluation.		
Signature		Date	19/07/2017

Please cut and paste additional co-author panels here as required.

Statement of Authorship

Title of Paper	Formation control based on local optima and multi-object mapping
Publication Status	□ Published □ Accepted for Publication ■ Submitted for Publication □ Unpublished and Unsubmitted w ork w ritten in manuscript style
Publication Details	Yu, H., Shi, P., Lim C. C., & Wu, L. (2017). Formation control based on local optima and multi-object mapping, submitted and under review.

Principal Author

Name of Principal Author (Candidate)	Hongjun Yu		
Contribution to the Paper	Conducted theoretical derivation, performed practical analysis on all numerical examples, interpreted data, wrote manuscript and acted as corresponding author.		
Overall percentage (%)	90%		
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.		
Signature		Date	19/07/2017

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

i. the candidate's stated contribution to the publication is accurate (as detailed above);

ii. permission is granted for the candidate in include the publication in the thesis; and

iii. the sum of all co-author contributions is equal to 100% less the candidate's stated contribution.

Name of Co-Author	Peng Shi		
Contribution to the Paper	Supervised development of work, helped in data interpretation and manuscript evaluation.		
Signature		Date	19/07/2017

Name of Co-Author	Cheng-Chew Lim		
Contribution to the Paper	Supervised development of work, helped in data interpretation and manuscript evaluation.		
Signature		Date	19/07/2017

Name of Co-Author	Ligang Wu		
Contribution to the Paper	Supervised development of work, helped in data interpretation and manuscript evaluation.		
Signature		Date	19/07/2017

Please cut and paste additional co-author panels here as required.

Statement of Authorship

Title of Paper	Formation control for multi-robot systems with collision avoidance		
Publication Status	□ Published ■ Submitted for Publication	□ Accepted for Publication □ Unpublished and Unsubmitted w ork w ritten in manuscript style	
Publication Details	Yu, H., Shi, P., Lim C. C., & Wang, D. (2017). Formation control for multi-robot systems with collision avoidance, submitted and under review.		

Principal Author

Name of Principal Author (Candidate)	Hongjun Yu		
Contribution to the Paper	Conducted theoretical derivation, performed practical analysis on all numerical examples, interpreted data, wrote manuscript and acted as corresponding author.		
Overall percentage (%)	90%		
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.		
Signature		Date	19/07/2017

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

i. the candidate's stated contribution to the publication is accurate (as detailed above);

ii. permission is granted for the candidate in include the publication in the thesis; and

iii. the sum of all co-author contributions is equal to 100% less the candidate's stated contribution.

Name of Co-Author	Peng Shi		
Contribution to the Paper	Supervised development of work, helped in data interpretation and manuscript evaluation.		
Signature		Date	19/07/2017

Name of Co-Author	Cheng-Chew Lim		
Contribution to the Paper	Supervised development of work, helped in data interpretation and manuscript evaluation.		
Signature		Date	19/07/2017

Name of Co-Author	Dongzhe Wang		
Contribution to the Paper	Involved in acquisition of experimental data and hardware setup.		
Signature		Date	19/07/2017

Please cut and paste additional co-author panels here as required.

Statement of Authorship

Title of Paper	Formation control of arbitrary shape with no communication
Publication Status	■ Published □ Accepted for Publication □ Submitted for Publication □ Unpublished and Unsubmitted w ork w ritten in manuscript style
Publication Details	Yu, H., Shi, P., & Lim, C. C.. Formation control of arbitrary shape with no communication. In Intelligent Control and Automation (WCICA), 2016 12th World Congress on (pp. 356-359). IEEE. 12-15 June 2016. Guilin, China.

Principal Author

Name of Principal Author (Candidate)	Hongjun Yu		
Contribution to the Paper	Conducted theoretical derivation, performed practical analysis on all numerical examples, interpreted data, wrote manuscript and acted as corresponding author.		
Overall percentage (%)	90%		
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.		
Signature		Date	19/07/2017

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

i. the candidate's stated contribution to the publication is accurate (as detailed above);

ii. permission is granted for the candidate in include the publication in the thesis; and

iii. the sum of all co-author contributions is equal to 100% less the candidate's stated contribution.

Name of Co-Author	Peng Shi		
Contribution to the Paper	Supervised development of work, helped in data interpretation and manuscript evaluation.		
Signature		Date	19/07/2017

Name of Co-Author	Cheng-Chew Lim		
Contribution to the Paper	Supervised development of work, helped in data interpretation and manuscript evaluation.		
Signature		Date	19/07/2017

Please cut and paste additional co-author panels here as required.

Statement of Authorship

Title of Paper	Matching based formation control and analysis of large-scale multi-agent systems
Publication Status	■ Published □ Accepted for Publication □ Submitted for Publication □ Unpublished and Unsubmitted w ork w ritten in manuscript style
Publication Details	Yu, H., Shi, P., & Lim, C. C.. Matching based formation control and analysis of large-scale multi-agent systems. In Systems, Man, and Cybernetics (SMC), 2015 IEEE International Conference on (pp. 173-178). IEEE. 9-12 October 2015. Kowloon, Hong Kong, China.

Principal Author

Name of Principal Author (Candidate)	Hongjun Yu		
Contribution to the Paper	Conducted theoretical derivation, performed practical analysis on all numerical examples, interpreted data, wrote manuscript and acted as corresponding author.		
Overall percentage (%)	90%		
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.		
Signature		Date	19/07/2017

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

i. the candidate's stated contribution to the publication is accurate (as detailed above);

ii. permission is granted for the candidate in include the publication in the thesis; and

iii. the sum of all co-author contributions is equal to 100% less the candidate's stated contribution.

Name of Co-Author	Peng Shi		
Contribution to the Paper	Supervised development of work, helped in data interpretation and manuscript evaluation.		
Signature		Date	19/07/2017

Name of Co-Author	Cheng-Chew Lim		
Contribution to the Paper	Supervised development of work, helped in data interpretation and manuscript evaluation.		
Signature		Date	19/07/2017

Please cut and paste additional co-author panels here as required.

Statement of Authorship

Title of Paper	Probability-triggered formation control with limited sensing range
Publication Status	□ Published □ Accepted for Publication ■ Submitted for Publication □ Unpublished and Unsubmitted w ork w ritten in manuscript style
Publication Details	Yu, H., Shi, P., & Lim C. C. (2017). Probability-triggered formation control with limited sensing range, submitted and under review.

Principal Author

Name of Principal Author (Candidate)	Hongjun Yu		
Contribution to the Paper	Conducted theoretical derivation, performed practical analysis on all numerical examples, interpreted data, wrote manuscript and acted as corresponding author.		
Overall percentage (%)	90%		
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.		
Signature		Date	19/07/2017

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

i. the candidate's stated contribution to the publication is accurate (as detailed above);

ii. permission is granted for the candidate in include the publication in the thesis; and

iii. the sum of all co-author contributions is equal to 100% less the candidate's stated contribution.

Name of Co-Author	Peng Shi		
Contribution to the Paper	Supervised development of work, helped in data interpretation and manuscript evaluation.		
Signature		Date	19/07/2017

Name of Co-Author	Cheng-Chew Lim		
Contribution to the Paper	Supervised development of work, helped in data interpretation and manuscript evaluation.		
Signature		Date	19/07/2017

Please cut and paste additional co-author panels here as required.

Statement of Authorship

Title of Paper	Robot formation control in stealth mode with scalable team size
Publication Status	■ Published □ Accepted for Publication □ Submitted for Publication □ Unpublished and Unsubmitted w ork w ritten in manuscript style
Publication Details	Yu, H., Shi, P., & Lim, C. C. (2016). Robot formation control in stealth mode with scalable team size. International Journal of Control, 89(11), 2155- 2168.

Principal Author

Name of Principal Author (Candidate)	Hongjun Yu		
Contribution to the Paper	Conducted theoretical derivation, performed practical analysis on all numerical examples, interpreted data, wrote manuscript and acted as corresponding author.		
Overall percentage (%)	90%		
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.		
Signature		Date	19/07/2017

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

i. the candidate's stated contribution to the publication is accurate (as detailed above);

ii. permission is granted for the candidate in include the publication in the thesis; and

iii. the sum of all co-author contributions is equal to 100% less the candidate's stated contribution.

Name of Co-Author	Peng Shi		
Contribution to the Paper	Supervised development of work, helped in data interpretation and manuscript evaluation.		
Signature		Date	19/07/2017

Name of Co-Author	Cheng-Chew Lim		
Contribution to the Paper	Supervised development of work, helped in data interpretation and manuscript evaluation.		
Signature		Date	19/07/2017

Please cut and paste additional co-author panels here as required.

Statement of Authorship

Title of Paper	Scalable formation control in stealth with limited sensing range
Publication Status	■ Published □ Accepted for Publication □ Submitted for Publication □ Unpublished and Unsubmitted w ork w ritten in manuscript style
Publication Details	Yu, H., Shi, P., & Lim, C. C. (2017). Scalable formation control in stealth with limited sensing range. International Journal of Robust and Nonlinear Control, 27(3), 410-433.

Principal Author

Name of Principal Author (Candidate)	Hongjun Yu		
Contribution to the Paper	Conducted theoretical derivation, performed practical analysis on all numerical examples, interpreted data, wrote manuscript and acted as corresponding author.		
Overall percentage (%)	90%		
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.		
Signature		Date	19/07/2017

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

i. the candidate's stated contribution to the publication is accurate (as detailed above);

ii. permission is granted for the candidate in include the publication in the thesis; and

iii. the sum of all co-author contributions is equal to 100% less the candidate's stated contribution.

Name of Co-Author	Peng Shi		
Contribution to the Paper	Supervised development of work, helped in data interpretation and manuscript evaluation.		
Signature		Date	19/07/2017

Name of Co-Author	Cheng-Chew Lim		
Contribution to the Paper	Supervised development of work, helped in data interpretation and manuscript evaluation.		
Signature		Date	19/07/2017

Please cut and paste additional co-author panels here as required.

Statement of Authorship

Title of Paper	Steady formation analysis on multi-robot systems
Publication Status	■ Published ☐ Accepted for Publication ☐ Submitted for Publication ☐ Unpublished and Unsubmitted w ork w ritten in manuscript style
Publication Details	Liu, Y., Yu, H., Lim, C. C., & Shi, P. (2017). Steady formation analysis on multi-robot systems. Journal of Control and Decision, 4(1), 12-31.

Principal Author

Name of Principal Author (Candidate)	Hongjun Yu		
Contribution to the Paper	Conducted theoretical derivation, performed practical analysis on all numerical examples, interpreted data.		
Overall percentage (%)	40%		
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.		
Signature		Date	19/07/2017

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

i. the candidate's stated contribution to the publication is accurate (as detailed above);

ii. permission is granted for the candidate in include the publication in the thesis; and

iii. the sum of all co-author contributions is equal to 100% less the candidate's stated contribution.

Name of Co-Author	Yutong Liu		
Contribution to the Paper	Conducted theoretical derivation, performed practical analysis on all numerical examples, interpreted data, wrote manuscript and acted as corresponding author.		
Signature		Date	19/07/2017

Name of Co-Author	Peng Shi		
Contribution to the Paper	Supervised development of work, helped in data interpretation and manuscript evaluation.		
Signature		Date	19/07/2017

Name of Co-Author	Cheng-Chew Lim		
Contribution to the Paper	Supervised development of work, helped in data interpretation and manuscript evaluation.		
Signature		Date	19/07/2017

Please cut and paste additional co-author panels here as required.