THE UNIVERSITY OF ADELAIDE

DOCTORAL THESIS

Formation Control and Reconfiguration Strategy of Multi-Agent Systems

Author:
Yutong LIU

Supervisor:
Prof. Peng SHI
Prof. Cheng-Chew LIM

A thesis submitted in fulfillment of the requirements for the degree of Doctor of Philosophy

in the

Systems & Control Group
School of Electrical & Electronic Engineering

March 5, 2020

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 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 University's 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: 03/05/2020

Abstract

Multi-agent systems consist of multiple agents, which detect and interact with their local environments. The formation control strategy is studied to drive multi-agent systems to predefined formations. The process is important because the objective formation is designed such that the group achieves more than the sum of its individuals.

In this thesis, we consider formation control strategies and reconfiguration strategy for multi-agent systems. The main research contents are as follows.

A formation control scheme is proposed for a group of elliptical agents to achieve a predefined formation. The agents are assumed to have the same dynamics, and communication among the agents limited. The desired formation is realized based on the reference formation and the mapping decision. In the controller design, searching algorithms for both cases of minimum distance and tangents are established for each agent and its neighbors. In order to avoid collision, an optimal path planning algorithm based on collision angles, and a self-center-based rotation algorithm are also proposed. Moreover, randomized method is used to provide the optimal mapping decision for the underlying system.

To optimize the former formation control scheme, an adaptive formation control strategy is developed. The multiple elliptical agents can form a predefined formation in any 2D space. The controller is based on the neighborhood of each agent and the optimal mapping decision for the whole group. The collision-free algorithm is built based on direction and distance of avoidance group of each agent. The controller for each agent is adaptive based on the number of elements in its avoidance group, the minimum distance it has and its desired moving distance. The proposed adaptive mapping scheme calculates the repetition rate of optimal mappings in screening group of mapping decisions. The new optimal mapping is constructed by the fixed repeating elements in former mappings and the reorganized elements which are not the same in each optimal mappings based on the screening group.

An event-triggered probability-driven control scheme is also investigated for a group of elliptical agents to achieve a predefined formation. The agents are assumed to have the same dynamics, and the control law for each agent is only updated at its event sequence based on its own minimum collision time and deviation time. The collision time of each agent is obtained based on the position and velocity of the others, and the deviation time is linked with the distance between its current position and desired position. The probability-driven controller is designed to prevent the stuck problem among agents. The stuck problem for the group means that when the distance between

agents is too close and their moving directions are crossed, the control input with deterministic direction will cause the agents not to move or to move slowly. To optimize the event-triggered probability-driven controller, a mapping-adaptive strategy and an angle-adaptive scheme are also developed. The mapping-adaptive strategy is used to find the optimal mapping to decrease the sum of the moving distance for the whole group, while the angle-adaptive scheme is employed to let the distance between any two elliptical agents is large enough to further ensure there is no collision existed during execution.

Reconfiguration strategy is considered for multiple predefined formations. A two-stage reconfiguration strategy is proposed for a group of agents to find its special formation, which can be seen as transition of the predefined formations, during idle time in order to minimize the reconfiguration time. The basic reconfiguration strategy combines with a random mapping algorithm to find optimal special formation. To meet the practical requirements, agents are modeled as circles or ellipses. The anti-overlapping strategies are built to construct the achievable special formation based on the geometric properties of circle and ellipse.

Keywords: Multi-agent systems, formation control, collision avoidance, random mapping algorithm, event-triggered, reconfiguration strategy

Acknowledgements

I would like to dedicate my thesis to all those who have offered me tremendous assistance during the five years in School of Electric & Electronic Engineering, the University of Adelaide.

First of all, my heartiest thanks flow to my supervisor, Professor Peng Shi and Professor Cheng-Chew Lim, for their helpful guidance, valuable suggestions and constant encouragement both in my study and in my life. Their profound insight and accurateness about my thesis taught me so much that they are engraved on my heart. They provided me with beneficial help and offered me precious comments during the whole process of my writing, without which the thesis would not be what it is now. They have offered me valuable suggestions in the academic studies. I would also like to thank Mrs Shi for caring my life during my study in Adelaide and treating me as her own child.

I owe many thanks to School of Electrical & Electronic Engineering and Faculty of Engineering, Computer & Mathematical Sciences of the University of Adelaide for providing all necessary supports.

My thanks would go to my beloved family for their loving considerations and great confidence in me all through these years and their supporting without a word of complaint. I also owe my sincere gratitude to my friends and colleagues, Hongjun Yu, Wen Xing, Bing Yan, Yueyuan Zhang, Jie Han, Xin Yuan, Qianhui Wan, Xin Zhang, and Yang Zhao, who have given me their help and their time in listening to me and helping me work out my problems during the difficult course of the thesis. Last but not the least, I thank my boyfriend, Xinyu Zhang for his love and patience, and encouragement during my study.

Publications

Journal publications

[1] **Y. Liu**, H. Yu, P. Shi , & C. C. Lim. Formation control and collision avoidance for a class of multi-agent systems. Journal of the Franklin Institute, 356(10), 5395-5420, 2019.

[2] **Y. Liu**, P. Shi, & C. C. Lim. Collision-free formation control for multi-agent systems with dynamic mapping. IEEE Transactions on Circuits and Systems II: Express Briefs, 2019.

[3]**Y. Liu**, P. Shi, H. Yu, & C. C. Lim. Event-triggered probability-driven adaptive formation control for multiple elliptical agents. IEEE Transactions on Systems, Man and Cybernetics: Systems, revised version under review.

[4]**Y. Liu**, P. Shi, C. C. Lim, & H. Yu. Two-stage reconfiguration strategy for multi-agent systems. IEEE Transactions on Circuits and Systems I: Regular Papers, under review.

Contents

List of Figures

List of Tables

List of Symbols

Symbols in Chapter 2 and Chapter 3

E_i the ith agent
$p_i = (x_i, y_i)$ the coordinate of point p_i
$p_i^c = (x_{i0}, y_{i0})$ the center coordinate of agent E_i
a_i the long-axis of agent E_i
b_i the long-axis of agent E_i
ϕ_i the heading angle of agent E_i
θ_i the inner angle of agent E_i
Θ_i the set of chosen inner angle θ_i and changing angle based on θ_i of agent E_i
u_i the position control force of agent E_i
u_{ϕ_i} the control force of the heading angle of agent E_i
R_{sen} the sensing range of elliptical agents
R_{avo} the avoidance range of elliptical agents
$H_{E_i}(R_{sen})$ the neighborhood of agent E_i
$A_{E_i}(R_{avo})$ the avoidance group of agent E_i
F the predefined formation for the whole group
$\bar{F}_{f_i}$ the set of displacements of f_i
g_i the desired position of agent E_i
R^k the random mapping pool in the kth iteration
M^k the mapping pool in the kth iteration
r_{op}^{k-1} the optimal mapping in the $(k-1)$th iteration
L^k the sum of corresponding distances based on M^k in the kth iteration
l_2^{ij}, l_3^{ij} the intersect tangents between E_i and E_j
ψ_{ij} the collision angle between l_2^{ij} and l_3^{ij}
Ψ_{ij} the set of ψ_{ij}
m_{ij} the midline between l_2^{ij} and l_3^{ij}
M_{ij} the set of m_{ij}
φ_{ij} the angle between u_i and m_{ij}
Ω_{ij} the set of φ_{ij}
u_i^d the update position control force of E_i
g_i^d the update desired position of E_i
d_{il}^{min} the minimum distance between E_i and E_l
$\xi_i(k)$ the adaptive parameter of E_i, set in (3.12)

$R_f(k)$ the screening group in the kt iteration
$R_r(k)$ the repeating mappings set in $R_f(k)$

Symbols in Chapter 4

t^i the triggering time sequence of agent E_i
$C_i(t_k^i)$ the collision set of agent E_i at t_k^i
$\varphi_i j(t_k^i)$ the angle between control input of E_i and E_j at t_k^i
$\triangle v_i j(t_k^i)$ the speed difference of E_i and E_j at t_k^i
$d_i j(t_k^i)$ the distance between E_i and E_j at t_k^i
$m_i j(t_k^i)$ the slope of $\triangle v_i j(t_k^i)$ at t_k^i
$d_{ij}^{col}(t_k^i)$ the collision distance between E_i and E_j at t_k^i
$t_{ij}^{col}(t_k^i)$ the collision time of E_i and E_j at t_k^i
$D_i^{col}(t_k^i)$ the collision distance set of E_i at t_k^i
$t_i^{col}(t_k^i)$ the collision time set of E_i at t_k^i
$\tau_i^{col}(t_k^i)$ the minimum value of $t_i^{col}(t_k^i)$ at t_k^i
$B_i(t_k^i)$ the coefficient matrix of the deviation circle of E_i at t_k^i
$d_i^{dev}(t_k^i)$ the deviation distance of agent E_i at t_k^i
$t_i^{dev}(t_k^i)$ the deviation time of agent E_i at t_k^i
$Timer(t_k^i)$ the event-triggered timer of agent E_i at t_k^i
$R_i(t_k^i)$ the rotation matrix of agent E_i at t_k^i
$\theta_i(t_k^i)$ the probability rotation angle of agent E_i at t_k^i
$L(t_k^i)$ the sum of distance to reach desired positions at t_k^i
$\bar{u}_{\phi_i}(t_k^i)$ the heading angle control input of agent E_i at t_k^i

Symbols in Chapter 5

F the predefined formation set
$M^s(k)$ the mapping relations in the kth iteration
M_0^s the set of the initial mapping relations of the predefined formations
M_{op} the optimal mapping set
$p_j(a)$ the position for each dot agent in special formation
q_j^i the probability of occurrence of the agent E_j in predefined formation F^i
$P(c)$ the optimal special formation for circular agents

D^f the reference distance set of circular agents
D^c the actual distance set of circular agents
$\bar{D}$ the difference between D^f and D^c
D^j the coincidence distance set of circular agent E_j
D_p^j the projected coincidence distance onto x-axis set of agent E_j
θ_{jl} the projected angle of $d_l \in D^j$
λ_j the moving vector of agent E_j
$P(e)$ the optimal special formation for elliptical agents
m_{jl} the slope of line between intersection points between elliptical agents E_j and E_l

1 Introduction

1.1 Motivations

This thesis focuses on the formation control strategies for a group of elliptical agents and the two-stage reconfiguration strategy for a group of agents. These elliptical agents are moving in a physical or virtual formation, sensing and interacting with local environments. The formation control technologies for multiple agents are employed in various engineering fields to reduce the system cost, increase the robustness and efficiency of the system, and provide redundancy, reconfiguration capability and flexibility. However, formation control strategy will create many challenges for the systems with weakening each agent's processing capability and increasing the complexity of whole system. The design on agents is essentially based on egocentrism, as studied in [1]–[4], but agents free will be restricted by localisation from limited sensing range and communication capacities. Switching formations should be owned by agents groups to reduce moving consumption, decrease collision probability, and improve tasks completion rate, etc. For multi-agent systems, collective objectives are still existed, thus, control strategies design remains to deal with these goals. Tasks changing during whole groups is another problem that agents face. Reconfiguration ability is an essential feature of multi-agent system because it links to the problem of how many types of formations can be formed and transformed. In order to build the reconfiguration ability of multi-agent systems, reconfiguration strategies should be employed to drive the systems to construct different shapes of formation.

Conventional formation control strategies are insufficient in dealing with the shape of individual agent. In vast of literatures, agents are always considered as dots, circles, or rectangles [5]–[8]. However, many potential agents in practice have long and narrow shapes. Given the same length and width, the ellipse has a smaller area and smooth curve than that of rectangular shape. It is more appropriate to choose the ellipse shape to define a real agent. If we only consider point-shape, disk-shape or rectangular-shape for actual agents, collision problems among agents will be out of consideration, or many available spaces, which can be used to plan routes and prevent possible collisions among agents, will be wasted. Simultaneously, mapping relationships among multiple agents are important to research to release computational and communication burden and reduce operation cost. For individual agent, it need to make decisions wisely with its neighbors based on sensing capability and communication topology. For reconfiguration researches on multi-agent systems, adaptive mappings for multiple agents and the geometric shape of individual agent are lack of consideration. These control schemes could be developed to enhance the operation efficiency and improve the execution time.

The thesis aims to construct formation control strategies and two-stage reconfiguration strategy for multiple agents. This is achieved by developing schemes to manipulate the agents motives under the information from sensors and communication networks. The schemes are dominantly agent-based and they are well suited to provide solutions in localised and decentralised systems. Inter-agent communication which is only transfer some simple information such as individual identities, mapping relationship and positions of agents, is assumed to be established on local wireless network. Agents are abstract as ellipses to meet the practical requirements. Collision avoidance schemes among agents are improved to make sure agents can reach their desired goals well. The network is decentralized because the agents perform different position tasks. The agents are localised since they work in a large space, and the centralized communication network has a heavy cost burden for individual agent. Our formation control strategies and two-stage reconfiguration schemes can execute formation and reconfiguration goals perfectly.

Contribution from our work in this thesis is as follows:

1 Develop a new formation control scheme to drive a group of elliptical agents to a predefined formation with restricted communication and limited sensing capability. The controller of each agent is established based on the midpoint derived from their neighborhood. Random mapping algorithm is constructed by the sum moving distances from current positions of individual elliptical agents to the desired positions. The desired formation of the whole group does not match the specific positions of the predefined formation. It is obtained by the displacements from the predefined formation. The collision among elliptical agents can be avoided by choosing optimal path and removing obstacle angles. A self-center-based rotation algorithm is also proposed to guarantee collision avoidance when two agents approach to each other.

2 Investigate an adaptive collision-free control scheme and mapping rule based on the first developed formation control scheme. The communication among agents is needed to exchange mapping information and agent identities. The control input for each agent which is moving though the possible collision area, is adaptive based on the collision group of the agent and the distance between current position and desired moving position of the agent. An adaptive mapping algorithm is proposed to release the computational burden of random mapping algorithm.

3 Build an event-triggered probability-driven adaptive formation control strategy for multiple elliptical agents. Each agent has its own event sequence based on the minimum collision time and the deviation time calculated by itself. Agents only need to receive the state and velocity information in accordance with their own event sequence. Probability-driven controller is established to prevent the stuck problem among

agents, which may happen when two or more elliptical agents are too close to each other. The adaptive mapping scheme is employed to find the optimal mapping among the agents to reduce the moving distance of the whole system.

4 Propose a two-stage reconfiguration strategy based on dot agents during idle time with an adaptive random mapping algorithm, which is constructed based on the minimum expected moving distance between the current positions of the group of agents and each predefined formation. To meet the practical requirements, the two-stage reconfiguration schemes are improved due to the circular shapes and elliptical shapes of the agents, to deal with the overlapping problem happens among agents.

1.2 Research background

Nowadays, the traditional control theory on the control of an individual system has gradually matured. With the rapid development of computer, communication and sensor technologies, networked systems and multi-agent system with multiple nodes and communication network connections have been increasingly studied. The control technology of multi-agent systems have begun to be applied in large-scale applications in practical scenarios. In the practical applications, individual equipment cannot achieve the control goals with high efficiency and low cost. To improve operational efficiency, increase running accuracy, reduce operating costs and decrease maintenance difficulty, multiple small devices with low cost, simple structure, and easy assembly and maintenance are employed to work together to achieve the control goal to replace an individual complex agent. Compared with single-agent systems, multi-agent systems have the following advantages: 1) co-operation among agents can greatly expand the task execution ability of automation device. Based on the extension of the task execution ability, multi-agent systems can accomplish many complex tasks, which are difficult to achieve by an individual agent; 2) multi-agent systems have lower energy cost, and are easier to manufacture install and maintain, which will lead to a better economic efficiency beneficial; 3) multi-agent systems have better performance and higher efficiency; 4) lager system redundancy can be designed for multi-agent systems, which can achieve better robustness and fault tolerance design than an individual agent.

The multi-agent systems can be seen as "society of agents", which means a set of agents that interact together to coordinate their behaviour and often cooperate to achieve some collective goal, which is difficult or impossible for an individual agent or a monolithic system to solve. Also an agent in the multi-agent system is considered as a micro-system. In the systems, agents change their information to their neighbours or leaders by communication links, which can constitute a network called communication topology. The inspiration for collaborative control of distributed multi-agent systems comes from nature. Many biome systems have complex and interesting cluster behaviors. Animal groups exhibit a superior ability to manage a variety of challenging tasks, from foraging to migration to predator evasion. For example, a group of fish will cooperate together to prevent other fish attacks, a flock of birds forage and probe for food, and a group of wildebeests move together to migrate to a lush grassland to ensure race continues, and etc.. Individual animal in these biological flocks cannot perceive and process the global information of nature and its own cluster. They can only rely on its own perception system to obtain the status of their local neighbors and interact with them to achieve global behavior goals. Communication network among these animals is necessary to ensure the information exchange among animal groups. Social interactions among individuals are beneficial to the whole groups. The design method and control goal of distributed multi-agent cooperative control is learn from such behaviors. The design concept of distributed multi-agent control strategy is to design appropriate control schemes to complete complex global tasks with only local information interaction.

Formation control is one of the main applications of multi-agent systems. Agents in the multi-agent systems are required to follow a predefined trajectory while maintaining a desired geometry pattern. Moving in group has abundant advantages, such as, reduction of the system cost, increase of the robustness and efficiency of the system during providing redundancy, reconfiguration ability and structure flexibility for the system. There are five main control problems in formation control: formation generation, formation maintain, formation switching, obstacle avoidance and adaptive. The formation maintaining and formation switching attract more of my concern. How much behaviour and information of agents has poses direct influence on performance of the system, in terms of energy consumption and executing time from initiation to finishing a task. However, global information cannot always be gotten by each agent in the system or could lead high energy consumption. Therefore, researches have been done to improve the performance under these circumstances.

1.2.1 The concept of multi-agent systems

Multi-agent system is a system consists of a group of agents. It can solve problems which are difficult for single agent through communication, consulting and cooperating among agents and environment. There are more advantages for using multi agents. Multiple agents cooperate with each other can complete the task beyond the scope of capacity of single agent which lead overall capacity of whole system better than single one. The concept of multi-agent systems is first brought up in computer science. To build a clear understanding of a multi-agent system, the definition of agent is necessary. In the domain of computer, an agent can be defined as "An agent is a computer system that is situated in some environment, and that is capable of autonomous action in this environment in order to meet its design objectives"[9].

There are two kinds of interactions in an agent. The first one is the interaction between an agent and its nearby environment, as illustrated in Figure 1.1. Agents can accept the input from environment and produce an action output to influence the nearby environment. This interaction is always a continuous process. In most cases, agents can only have partial control to their nearby environment, and influence the environment by its action output. Agents can influence the system in several ways. They either could produce some change in the current state of its environment which changes future interactions of other agents with it, or they change their own performance which changes interactions with neighbour agents who see these agents as a part of the environment. Therefore, agent can produce same action output in interaction with same environment which could lead different results, rendering it nondeterministic.

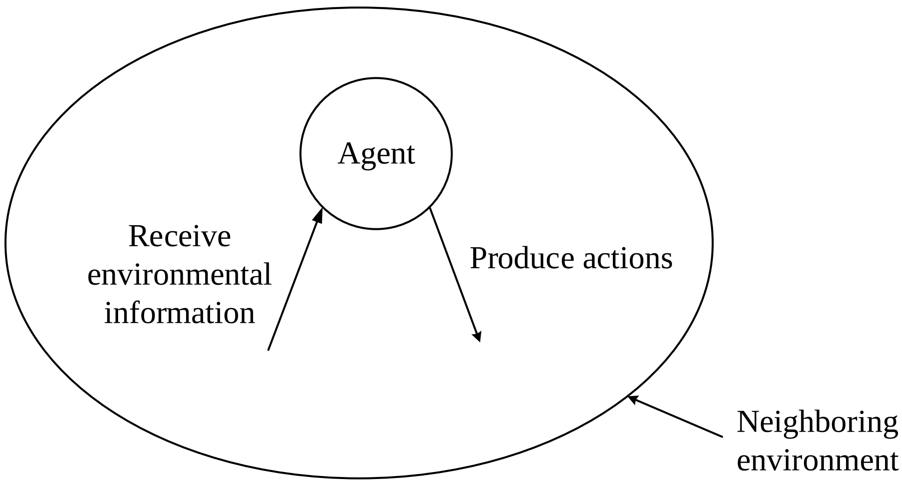

FIGURE 1.1: The interaction between an agent and its nearby environment

The other interaction is the interaction between an agent and its neighbour agents. This inter-agent interaction is possible because of communications among agents. Different from the interaction between an agent and its environment, an agent may not be able to directly influence the state of its neighbour agents. In multi-agent system, it is possible for some agents to grasp the global information due to their role playing in the whole system. There may also be cases where these agents can communicate with all of the rest of agents in the system. This feature is captured in my research where agents which are seen as leaders can share global information and communicate with all the rest of agents in the multi-agent system. Figure 1.2 shows the relationship among agents and their environment in a multi-agent system. It can be seen that the interaction among agents and their environment is complicated. The interaction among agents could be bidirectional, such as the interaction between agent 1 and agent 2, agent 1 and agent 3, agent 4 and agent 5, are bidirectional. Also the interaction among agents could be unidirectional. Sometimes, the environments of agents are cross, which means individual agent may be influenced when others influence the crossed environment based on their own action. This fact will cause dependent relationship among agents. In some cases, agents will not interact with environment.

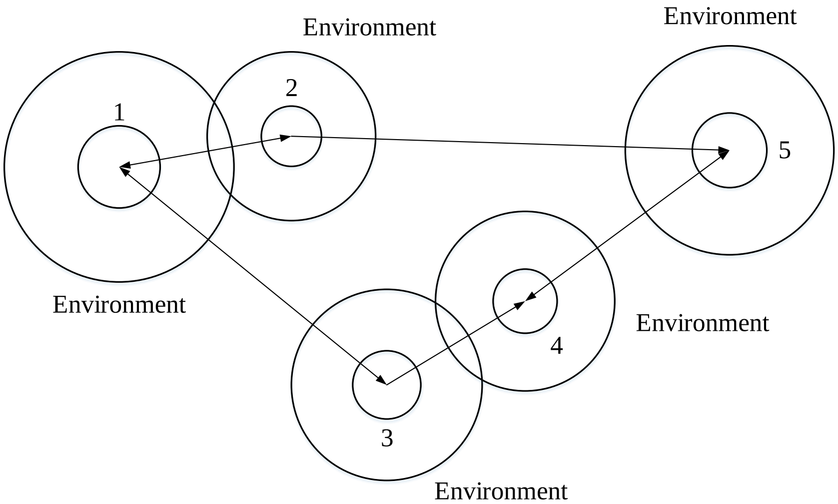

FIGURE 1.2: Inter-agent interaction

Organization of multi-agent systems means information and control relationship among agents, and the distributed mode of problem solving ability. As organization is the fundamental study of most research in multi-agent systems, it is important to organize agents effectively in order to cooperating agents and their environment. Agents could execute a prior organization to determine roles of each agent. The determined role gives the expectations about the agent's individual motion by describing its state, capabilities and gestures inside the multi-agent system. Then an agent achieves its organizational information of its role and executes tasks as the determined role. There are three kinds of organization of multi-agent systems[10]. The first one is centralized organization[11]–[13], which could be seen as a master-slaver hierarchical structure. In this mode, agents are divided into three categories, manager agent, functional agent and application agent by work range and function of agents. It is manager agent's responsibility to manage whole information, state and predefined task execution of the system. Functional agents could have capabilities to execute tasks and help manager agents to do system tasks without the influence from application agents. Application agents could process tasks in a specific field. Centralized structure can easily tolerate faults of some agents in the group. On the other hand, manager agents will be the weakness of the whole system both for computational and communication time requirements and its final fault will compromise the entire group[14]. Decentralized architecture[15]–[17] is the second mode of organization. Agents in this system are not controlled or directed by a single agent. They could get all information for other agents and their environment state due to each agent can communicate with other agents directly and immediately. According with these characteristics, agents in the system can plan their own trajectory and motives which could reflect autonomous and social of agents, the flexible and extensive of whole system could also be promoted. However, this could lead to uncontrollable and instability to the multi-agent system. The other structure is hybrid organization in which the responsibility and role of agent is dynamics[18], [19]. Interactions among agents are changing to cooperate more effectively to execute tasks. In my research, agents will organize in hybrid structure in which several leaders will be selected to tracks the predefined trajectory and other agents maintain a predefined direction and orientation with respect of the leaders. The leaders can change in accordance with varying needs of the system. In my works, decentralized organization for multiple agents is considered.

1.2.2 Applications of multi-agent systems

In recent years, models and related control theories of multi-agent systems have been applied in more and more engineering fields. In aerospace technology, a spacecraft can be regarded as an agent. The tasks, such as reducing system cost, improve system stability and function extension capability, can be achieved by developing coordinated attitude control and formation control of multiple spacecrafts. By using multi-agent technologies, multiple spacecraft systems, in which spacecrafts have simple structure and processors, can deal with collective targets that is difficult for a single spacecraft to process, which is studied in [20]–[23]. In the application of military technology, the use of multiple unmanned aerial vehicles(UAVs)[24]–[27] to conduct reconnaissance and combat, the use of multi-robots [28]–[31] for searching, rescuing, patrol, clearance of mines and etc, or the use of autonomous underwater vehicles(AUVs) [32]–[35] to cruise under the sea, can greatly improve the overall combat capability, enhance tasks completion and accuracy, and reduce casualty. In industrial production processes [36]–[39], the use of multiple robotic arms to perform complex tasks on the production line can often improve assembly accuracy and production efficiency. The research of multi-agent systems, as a new and comprehensive cross-cutting topic, has a wide range of applications and huge potential value, attracts scholars in various fields, and promotes the rapid development of related theories. Some applications are listed to specified the use of multi-agent technology in engineering fields.

1 **Coordination and Control of mobile robots**
The problem of how to control and coordinate of multiple autonomous robots has attracted much attention during the last few years[40]–[46]. This task can decomposed into five subtasks, formation of geometric pattern, alignment of each robot's orientation, coordination of the robots in the system, motion realization and stability of the formation in motion. Generally speaking, autonomous robots should maintain a desired formation to promote effectiveness of whole system. A final orientation should be planned that robots will come to their positions from various locations so that their final directions of motion are different. To achieve the specific goal of the whole system, algorithms and controllers of coordination and control of autonomous robots should be

designed and proposed[43]–[45]. Realization of motions is used to satisfy constraint conditions while the group of robots is moving though a trajectory. At last, to make formation robust and not easy to disband, stability of formation during group moving should be considered[45], [46].

2 **Control of Traffic and Transportation**
The distributed nature of traffic control topology makes it well suited for applying multi-agent technology[47]–[51]. Distributed processing and coordination of multi-agent technology could be well used to solve the traffic conditions which contain dramatic changes such as traffic accidents[47], [48]. For urban traffic networks, intersections or junctions can be seen as nodes of multi-agent systems while links represent streets, avenues, roads or any other infrastructure connecting them. Dynamic coupling graph and communication topology will be obtained to achieve predictive control[51]. The transit network-planning problems, timetabling and schedule synchronization, and the alignment problem for public transportation route can also be solved by using multi-agent technology[49], [50].

3 **Coordination Expert System**
For complex problems, single expert system cannot meet the requirements. Hence, multiple expert systems should be collaborated to achieve the collective goals[52], [53]. The multi-agent technique can be used to attain a coordination solution to multiple expert systems. Two expert systems can organically combine through a coordinated multi-agent system in a multi-agent system environment based on rules, and the coordination agreement is established to integrate diagnostic methods and improves the efficiency of fault diagnosis[53]. Expert systems can also be seen as nodes in multi-agent system to execute a specific task by cooperating and collision avoidance strategy should be employed to solve the conflict during processing tasks.

1.3 Literature review

1.3.1 Theoretical developments of formation control

In last decades, formation control became one of the leading research fields in multi-agent systems. Several methods have been proposed to deal with the formation control issues for a group of autonomous agents, including the leader-follower method, the behavioral method and the virtual structure method. The leader-follower method [54]–[58] treats a small group of agents in the system as leaders, and the rest can be seen as followers. The followers can find their desired position based on the positions of leaders and local information from the predetermined formation. This approach is easy to understand and implement. However it is hard to maintain a desired formation if followers are disturbed without any formation feedback. In [54], a non-singular terminal sliding-mode control and disturbance observer based control are designed for followers to achieve finite-time output consensus for the agents with mismatching disturbances of followers. A fixed-time consensus problem for a high-order leader-follower multi-agent system with external disturbances is studied in [55], in which a sliding manifold is built to make sure the tracking errors converge to zero in fix time. For followers whose relative states can not be measured, an observer-based distributed adaptive control scheme in [56], is constructed to ensure that all followers can follow their leader asymptotically. In [57], Two non-smooth leader-following formation protocols for nonidentical Lipschitz nonlinear multi-agent systems with directed communication network topologies are proposed for first-order systems and second-order systems. In [58], a leader-follower control scheme with time-varying unknown leader is investigated for the distributed tracking problem of nonlinear fractional-order multi-agent systems. Controllers are designed based on the basic independent behaviors of each agent in [59]–[63], and the behavior method can generate control law easily while each agent has multiple completive objectives, but it is difficult to design the local control rules and conduct stability analysis. In [59], a unified optimal control framework is constructed to integrate the formation control, trajectory tracking and collision avoidance in an obstacle-laden environment. A behavioural decentralized approach for multiple UAVs is proposed in [60]. Under this approach, individual UAV can fly though predefined waypoints, and avoid possible collisions with other UAVs in the group. A null-space-based behavioral method is developed in [61] to guarantee obstacle avoidance for multiple AUVs. In [62], a concept of escape angle is introduced into a decentralized behavior-based formation control algorithm for multi-robots systems to avoid obstacles. In [63], multiple missions control problem is solved as a behavioral control problem with the systematic procedure of null-space-based projection. In the virtual structure method [64]–[68], agents can achieve the desired formation with certain geometric shapes, which can be called as a virtual rigid. This approach can easily achieve the objective formation with high-precision trajectory tracking, but is difficult in implementing formation scaling and improving adaptability. In [64], the receding horizon tracking control of multiple unicycle-type robots is developed under coupled input constraint. The tracking position of the the follower is seen as a virtual structure point with a Frenet-Serret fram fixed on the leader. Shrinkable virtual structure is used in [65] to concur the traditional obstacle-avoidance problem in the virtual structure methodology. The virtual structure is employed in [66] to be regarded as a framework to plan and execute complex interleaved trajectories, which can hold a fixed relative formation, or transition between different formations. In [67], the virtual structure is introduced to build a formation control strategy for nonholonomic intelligent vehicles. And the algorithm of synchronized multiple spacecraft rotations based on consensus-based virtual structure is presented in [68], in which a behavioral consensus algorithm for virtual attitude control system, is presented to accomplish the attitude maneuver of the entire formation and guarantee a consistent attitude among the local virtual structure counterparts during the attitude maneuver.

Research works on formation control always use one or more of the above method under unlimited sensing capability and unrestricted communication. For limited sensing range study, in [69], platoon control with range-limited sensors is investigated to guarantee string stability and control performance within limited sensing. However, global broadcasts are used to attain string stability and this impose constraints to its applications. A simple controller is proposed in [70] for each agent in the system with same protocol and control law without consideration of potential field storage. To achieve local optimal mapping decision, two protocols for multi-object mapping are designed. In the work of [71], a bounded cooperative controller is designed with formation stabilization. In [72], potential functions are employed to achieve desired configuration while preventing collision with other agents, obstacles and the boundary of the work area. Agents in this research only have limited sensing and communication ranges. Restricted communication among agents is also discussed in [70], [73], [74] for minimizing energy consumption and improving adaptability to dynamic environmental changing. In [70], a novel displacement-based formation controller is proposed to drive multiple robots to their desired formation without communication based on their local in-range displacements from their own sensors. A formation tracking control scheme in [73] is constructed to maintain a given formation in nonomniscient constrained space. The role switching triggered is employed to enhance the efficiency of the whole algorithm. In [74], a distributed formation control and collision avoidance strategy is designed for a group of rectangular agents with limited communication ranges.

Fixed topology and switched topology in the area of formation are also researched in various articles. In [75], formation control for multi-agent systems, in which agents are controlled by unknown effect based on their states, are proposed on a fixed topology. A leader-follower consensus algorithm is given in [56] linked to relevant outputs of followers based on fixed topology. Consensus problems are solved in [76] by building a relation between network connectivity and performance of protocol in a switching topology. A linear consensus protocol is given in [77] to localize control strategies for second-order discrete-time agents based on switching topology. Time-varying formation is analyzed with directed switching topology in [78], in which formation is defined by specified piecewise continuously differentiable vectors. Practical systems of multiple unmanned aerial vehicles are studied in [79], [80] based on switching topology.

Collision avoidance problem among multiple agents during execution emerges accompanied by development of formation control strategy. Many researchers work in this field to ensure multi-agent systems run smoothly. In [81], a decentralized leader-follower formation controller is proposed with constrained position outputs within a given range of the unmanned surface vehicles. In [82], collision avoidance problem is addressed by a distributed model predictive control with a relatively non-conservative compatibility constraint. In their design, the terminal set is a positively invariant set with the tailored terminal controller and cost to guarantee the collision avoidance. Artificial potential functions, nonlinear tracking differentiators and a backstepping technique are employed into an observer-based cooperative time-varying formation maneuvering control in [83] to support collision avoidance among all autonomous surface vehicles without velocity measurements. Inner circular region and outer circular region are designed for each agent in [84] to construct collision avoidance controller. The potential function is chosen to meet the requirements formed by these two circular regions to prevent possible collision among the agents. A distributed containment control algorithm is developed based on a close-range omnidirectional relative distance sensor and the potential functions methodology to enhance the collision-avoidance capacity of all satellites in [85]. In [86], the model predictive control approach is employed into formation controller with delayed communication among networked mobile robots.

Adaptive technology is equipped in more and more formation control strategies to enable multi-agent systems to better adapt to the external environments and complete the predefined tasks, such as formation, tracking and maintaining former formation. In [43], an adaptive controller is developed by using the image information form an uncalibrated perspective camera, which is located at any position and orientation on the follower agent. An adaptive programming and internal mode principle are used in [87] to deal with leader-follower multi-agent systems. The states of followers are unknown and the leader of the whole system is a perturbed exosystem. Local relative information from agent's neighbors is employed to build an adaptive practical time-varying output formation tracking protocol in [88]. An adaptive fuzzy logic system is introduced to estimate the mismatched uncertainties of the agents. Fully adaptive practical time-varying output formation tracking issues of high-order nonlinear stochastic multi-agent systems with multiple leaders are addressed by this formation tracking problem. In [89], an adaptive self-organizing map neural network approach is proposed to balance the execution workload of multiple AUVs. In the system, formation is treated as a distributed leader-follow structure-like, however, leaders and followers are not strictly determined. An adaptive distributed control scheme is constructed in [90] to guide the leaders into the predefined formation in finite time, while an adaptive controller is designed to keep followers maintaining the property distance and orientation from their leaders with the absence of unavailable inputs of their leaders. Backstepping technology is applied in a distributed adaptive controller in [91] for a kind of networked systems, which are consisted by various nonlinear subsystems with unknown parameters and non-identical nonlinear dynamics.

Agents in the multi-agent systems are often expressed as dots, circles or rectangular [56], [70], [74], [92], [93]. However, many practical agents have a long and narrow shape. Given the same length and width, the ellipse has a smaller area and smooth curve than that of rectangular shape. It is more appropriate to choose the ellipse shape to define a real agent. In [94], [95], potential function and goal function are employed to drive a group of elliptical agents to track predefined trajectories. A novel algorithm based on Minkowski sums and linear approximations for real-time obstacle avoidance among elliptical agents is proposed in [96].

1.3.2 Theoretical developments of event-triggered formation control

Event-triggered control is a control method based on conditional sampling, which is also known as Lebesgue sampling [97]. The event-triggered time is determined by the designed conditions, which is different from the time-triggered method, that is controlled by the clock cycle. From the perspective of system resource utilization, the traditional control scheme applied time-triggered condition is conservative. For example, we can use the system state in previous moment instead of the system state in current moment at two adjacent sampling moments, when the amplitude of the change in the system signal is extremely gentle, which leads the system state at adjacent moments is almost the same. However, under the time-triggered sampling control framework, the setting of the sampling period always needs to consider the worst-case scenario. The time interval should be designed to adapt to the scene with the most drastic signal change. The prior design of the upper and lower limits of the sampling period takes into account the need to meet the performance index requirements of the system's prior design. The sampling period must be preset according to the signal processing frequency bandwidth that the system can carry. However, the pre-designed sampling period in this way is not most suitable in all time periods. Redundancy may occur during time intervals that do not require that much sampling operations, which will lead the waste of the communication and computing resources and the system energy. To address this issue, event-triggered control scheme is put forward and developed fast.

Event-triggered approach is studied from [97]. In this research, event-triggered control strategy is proposed for some first-order systems. The researcher also compare the control result from event-triggered controller with the closed loop variance and sampling rate from periodic sampling. The earliest research on distributed event-triggered control theory for multi-agent systems started in 2009. In [98], an event-driven strategies for multi-agent systems is developed. The event-triggered condition is constructed by the ratio of a certain measurement error with respect to the norm of a function of the state. Nowadays, research on multi-agent systems under even-triggered control has been a lot of literature. The general classifications are as follows.

Research on system dynamics for event-triggered schemes can be divided into study on first-order integrator networks [99]–[103], and study on second-order integrator networks [7], [104]–[108], general linear systems [109]–[114] and nonlinear systems [115]–[118]. In [100], event-triggered distributed subgradient algorithms are developed for first-order discrete multi-agent systems to address convex optimization problems. Event-triggered scheme for each agent is constructed based on its own state and its neighbors' states.

Agents update their status with a designed centralized event-triggered function. An event-based impulsive controller is proposed in [102] for the consensus problems for single-integrator multi-agent systems. The trigger condition is designed based on the states of the agents. The consensus problem for heterogeneous first-order multi-agent systems in [103]. The consensus problem for second integrator multi-agent systems is solved in [104]. The controller for each agent is built relied on the state measurement error among its neighboring agents. In [106], an event-triggered control scheme is investigated to study the consensus of multiple second-order multi-agent systems under a directed spanning tree, where data is sampled randomly to reduce the communication load. The consensus study under undirected topology for second integrator multi-agent systems is constructed in [108]. Edge event-triggered scheme is based on the information of the corresponding two neighboring agents. General linear multi-agent systems are studied in the following literature. In [110], a dynamic compensator for each agent is introduced into the consensus controller to achieve output consensus for heterogeneous linear multi-agent systems. A decentralized event-triggered containment control strategy is investigated in [113] for heterogeneous linear multi-agent systems based on neighboring information. Exponential consensus problem of general linear multi-agent systems is addressed in [114], in which agents can only receive their neighbors' information only at their own triggering time instants. The controller for nonlinear multi-agent systems is researched in [115], [117]. In [115], a piecewise continuous control protocol combined with event-triggering function is proposed, while an event-triggered sampling control scheme with limited communication capability is studied in [117].

Research on topology for event-triggered schemes can be divided into directed topology [99], [101], [104]–[107], [114], [117] and undirected topology [100], [102], [103], [108]–[112], [115]. Classify by triggered type, event-triggered schemes for multi-agent systems can be divided into two categories: static trigger condition[119]–[121] and dynamic trigger condition [122], [123]. Static trigger condition methods include state-dependent trigger condition [124]–[126], time-dependent trigger condition [127]–[129] and integral-type trigger condition [121], [130]. Based on the measurement error, event-triggered multi-agent systems can be classified into continuous detection [121], [131], [132] and period detection [111], [126], [130] in which it can divide into synchronize detection [133], [134] and asynchronous detection [135]–[137]. The event-triggered control strategies can be also divided into point-state measurement(absolute state information measurement) [137], [138] and edge-state measurement(relative state information measurement) [108], [133] relying on status information measurement.

As a novel sampling sampling method, there are still a lot of issues that can be studied in the research of event-triggered distributed control scheme. We can explore the design of various practical event-triggered functions, consider more complex control objectives, study the flocking and formation problems of multi-agent systems. More complex controllers can be used to combine with event-trigger scheme, such as various types of output feedback dynamic controllers, sliding mode controllers, nonlinear observers, backstepping controllers, and active disturbance rejection controllers. More complex system dynamics models and network communication topology models can be considered, such as systems with switching dynamics, connectivity maintenance of the topology graph, high-order systems and etc. We can also develop event-trigger control approach in the view of performance analysis of distributed system, such as considering the robust performance, H_∞ performance, distributed optimization, and etc. The security issue is also a potential research direction in the event-sampling scheme of multiple agents systems. All of the above-mentioned extensions are theoretical work worthy of study. Moreover, the applications of event-triggered distributed control strategy should be investigated which can be applied in the piratical scenarios. The control schemes for multiple spacecrafts, AUVs, UAVs, and autonomous surface vehicles should be studied for enhancing the efficiency of practical systems.

1.3.3 Reconfiguration strategy of multi-agent systems

Reconfiguration ability is an essential feature of multi-agent systems because it links to the problem of how many types of formations can be formed and transformed. In order to build the reconfiguration ability of multi-agent systems, reconfiguration strategies should be employed to drive the systems to construct different shapes of formation.

There is a lot of research on reconfiguration strategy for multiple agents [139]–[145]. In [139], a distributed cascade robust feedback control scheme is built to drive a group of unmanned air vehicles which are vertical takeoff and landing to achieve the specific formation and reconfigured formation. The whole system is based on dynamic undirected communication network. The method based on coalition game theory and flocking-based formation maintenance mechanism is developed to solve the reconfiguration problem among multiple robots when they encounter obstacles in unknown environment in [140]. The problem of unsymmetrical formation reconfiguration and docking of multiple spacecrafts is studied in [141]. The compound optimal control is employed to deal with this problem with the total control effort and docking time, which are linked to the total fuel usage and electronic resources. In [142], a symplectic penalty iteration algorithm is proposed to achieve the optimal control for a group of spacecrafts, who have loose construction, with minimum energy consumption. In the algorithm, penalty functions are introduced to make sure the collision avoidance among spacecrafts. A continuous/impulsive linear quadratic regulator is constructed in [143]. It is used to design an optimal control scheme, which combines continuous Lorentz force actuation and impulsive thrusting, for spacecraft formation reconfiguration. Two-stage reconfiguration strategy is studied in [144], [145] for multiple aircrafts. The special designated formation in [144] is obtained based on the reconfiguration energy consumption. The special designated formation in [145], on the other hand, has the feature that the expected value of the reconfiguration time based on acceleration controller is minimized.

Reconfiguration of multi-agent systems technology is introduced to many engineering fields. The power distributed systems employ the reconfiguration scheme to solve outage problems and execution faults [146], [147], minimize active power loss and maximize voltage magnitude [148], [149], enhance voltage stability and load balancing [150], [151]. Spacecraft swarms can use the formation reconfiguration strategy to increase reliability and survivability, reduce system cost and risks, enhance mission flexibility [141]–[143], [152]. By applying reconfiguration technique into AUV swarms and UAV swarms, systems can enhance mission reliability and adaptability to changing mission requirements and decrease production and maintenance costs, which will lead to leading to technological and economic benefits [139], [153]–[155]. To address the severe faults, avoid surrounded obstacles, execute target detection and enclosing, reconfiguration strategies should be developed in multi-robot systems [140], [156]–[158].

1.4 Structure of thesis

Based on the above research motivations, collision-free formation control algorithms for a group of elliptical agents and the reconfiguration strategy for multi-agent systems are studied in this thesis. For collision-free formation control algorithms, we propose time-triggered method and event-triggered method for multiple elliptical agents to meet the practical requirements. The two-stage reconfiguration strategy for multiple agents is developed to achieve formation transition. The rest of this thesis is organised as follows.

In Chapter 2, a control algorithm, which is used to drive a group of elliptical agents to a predefined formation based on a reference formation, is proposed. The new technique is developed to achieve the objective included searching algorithms for finding minimum distance and tangents between two elliptical agents, which are used by the control algorithm. Communication among the agents is limited, and only identities of each agent and the mapping decision for them are transmitted. Random mapping algorithm is also presented to obtain optimal mapping decision for the whole group, in which a reference mapping is employed to provide displacements. The optimal mapping in each execution term can be obtained based on the sum of distances from current positions of all elliptical agents to their desired positions. Collision avoidance algorithm based on collision angles between tangents, which are between agents, is used to prevent collision among agents. By judging whether there is a possibility of collision between individual agent and the agents in its avoidance group, individual agent will make decision if it will update its control input or not. The self-center-based rotation algorithm for each agent is designed to further improve the collision avoidance. The simulations of fixed mapping and random mapping algorithm are given to demonstrate the feasibility and effectiveness of the new control design scheme.

In Chapter 3, an adaptive formation control strategy, which is used to enable a group of elliptical agents to achieve predefined formation is established. The control input of each agent is based on the displacements between the agent and its neighboring agents. To build the collision-free control strategy, the avoidance group of each agent based on the avoidance range and minimum distances among elliptical agents are employed. The adaptive formation controller for individual agent is proposed based on its minimum distance from the others, distance difference between its current position and its desired position and the number of the agents in its avoidance group. An adaptive random mapping algorithm is proposed for obtaining the optimal mapping decision. Different from the random mapping algorithm in Chapter 2, the adaptive random mapping algorithm will choose the fixed elements in the former optimal mappings to be regards as the elements in the final optimal mapping instead of generating new mappings in each iteration. Simulation results show the feasibility and effectiveness of the novel control strategy.

In Chapter 4, an event-triggered control algorithm to drive a group of elliptical agents in order to achieve a predefined formation is investigated. The control input update for each agent is event-driven, depending on the minimum collision time and deviation time of each agent. Each individual agent has its own event sequence. It can receive the state and velocity information of the others at the time when an event is triggered. The minimum collision time for individual agent is calculated based on the position and velocity of its nearest agent, while its deviation time means the moving time of an agent leaving its destination if the agent moves along the current control direction. The probability-driven control law is developed to prevent the stuck problem. The stuck problem for the group means that when the distance between agents is too close and their moving directions are crossed, the control input with deterministic direction will lead the agents not to move or move slowly. The probability-driven controller for each elliptical agent can produce a velocity, which has a different orientation with its original one, to bring agent out of dilemma. Also, adaptive algorithms of mapping and angle rotation are proposed to enhance the performance of event-triggered control algorithm. Mapping is updated based on the minimum distance of distance to reach predefined formation. The rotation algorithm is employed to expand the minimum collision distance among agents. Simulation results of the event-triggered control algorithm and event-triggered adaptive control algorithm are given to demonstrate the feasibility and effectiveness of the new control design scheme.

In Chapter 5, a two-stage reconfiguration strategy for a group of agents is constructed. By applying the two-stage reconfiguration algorithm during idle time, it can shorten the expected reconfiguration time when the next command with formation changing is given. These agents are modeled as dots, circles and ellipses to gradually approach the practical applications. For dot agents, the two-stage reconfiguration strategy combined with the random mapping algorithm is proposed to find the special formation during idle time based on optimal mappings to predefined formation set. Agents can find their special formation by using the probability of each formation in the predefined formation set. The two-stage reconfiguration scheme is improved for circular agents and elliptical agents to deal with the overlapping problem which may appear in the special formation by using the two-stage reconfiguration strategy for dot agents. The simulations of the two-stage reconfiguration strategy are given to demonstrate the feasibility and effectiveness of the new reconfiguration strategy.

Chapter 6 summarises the research findings and concludes the thesis.

2 Collision-free formation control for multiple elliptical agents

2.1 Introduction

In this chapter, a control scheme is investigated for a group of elliptical agents to achieve a predefined formation. The agents are assumed to have the same dynamics, and communication among the agents are limited. The desired formation is realized based on the reference formation and the mapping decision. In the control design, searching algorithms for both cases of minimum distance and tangents are established for each agent and its neighbors. In order to avoid collision, an optimal path planning algorithm based on collision angles, and a self-center-based rotation algorithm are also proposed. Moreover, randomized method is used to provide the optimal mapping decision for the underlying system.

In the current work, we consider agents as ellipses and investigate the formation control for a group of agents. Agents are limited with their sensing capability and restricted communication capability. They are equipped with displacement sensors which can provide displacements between agents and their neighbouring agents. The only data communication is the transmission of identities of the individual agents and the mapping decision for them. Moreover, agents are assumed to have the same dynamics and play equal roles in the whole system, which is different with[159]–[161]. Different from the work in [94], [95], this chapter focuses on driving a group of agents to achieve a desired formation derived from the objective map. The objective map is set well in advance and serves as a reference. Agents only organise their formation based on the displacements in the objective map, but not the specific points in the predefined map. The obstacle avoidance algorithm established in this chapter is based on the optimal path planning by removing the obstacle angles. These angles can be obtained by clamping tangents of objective agents and their obstacle agents.

In this chapter, the main work is as follows. First, a new control scheme is proposed to drive a group of elliptical agents to a predefined formation. All agents are assumed to have the same form of control law and reference formation. Only restricted communication among agents is allowed, and they can send and receive identification numbers to and from other agents in the system. The controller of each agent is established based on the midpoint derived from their neighbourhood. Second, the predefined formation is based on the displacements, which are obtained though a reference mapping. Agents can find their optimal mapping decisions based on the random mapping algorithm. During each sampling interval, several possible mappings are generated and the sums of distances with corresponding agents under each possible mapping decision are calculated to be compared with the others. The shortest one will be chosen to be the optimal formation in the corresponding interval. Third, the collision among elliptical agents can be avoided by choosing optimal path and removing obstacle angles. A self-center-based rotation algorithm is also proposed to guarantee collision avoidance when two agents approach to each other.

Notation. Throughout this chapter, $\mathbf{R}^m$ is an mth dimensional space of real numbers. For operations defined on groups, $A = \{a_1 \quad a_2 \quad \ldots\}$, $B = \{b_1 \quad b_2 \quad \ldots\}$, we have $C = A \cup B = \{c_1 \quad c_2 \quad \ldots\}$, where $c \in A$ or $c \in B$ for any $c \in C$. $a!$ is factorial of a, which represents $a \times (a-1) \times \ldots \times 2 \times 1$. The symbol $|\cdot|$ represents the length of a vector. For any two points c and d, $\overrightarrow{(c,d)}$ represents the vector between c and d. For two vectors e and f, $< e, f >$ represents the angle between e and f. Matrices are assumed to be compatible for algebraic operations. If dimensions of matrices are not explicitly stated, they are assumed to be compatible for algebraic operations.

2.2 Preliminaries

In our formation control problem, the agents are described as ellipses, and the relevant formulas will be given.

2.2.1 Elliptical formula of agent shape

Consider the ith elliptical agent $E_i \in \mathbf{R}^2$ whose heading angle is $\phi_i \in \mathbf{R}^2$, centered at (x_{i0}, y_{i0}), the elliptical representation is

$$E : [x, y, 1] \begin{bmatrix} \frac{\cos^2\phi_i}{a_i^2} + \frac{\sin^2\phi_i}{b_i^2} & \frac{\sin 2\phi_i}{a_i^2} & \frac{2A_1\cos\phi_i}{a_i^2} \\ -\frac{\sin 2\phi_i}{b_i^2} & \frac{\sin^2\phi_i}{a_i^2} + \frac{\cos^2\phi_i}{b_i^2} & \frac{2A_1\sin\phi_i}{a_i^2} \\ -\frac{2A_2\sin\phi_i}{b_i^2} & \frac{2A_2\cos\phi_i}{b_i^2} & \frac{A_1^2}{a_i^2} + \frac{A_2^2}{b_i^2} - 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = 0, \quad (2.1)$$

where, x and y are horizontal axis and vertical axis for the points on the ellipse, respectively. The semi-major axis of the E_i is represented by a_i, and b_i is the semi-minor axis with

$$A_1 = x_{i0}\cos\phi_i + y_{i0}\sin\phi_i,$$
$$A_2 = -x_{i0}\sin\phi_i + y_{i0}\cos\phi_i.$$

The set of points on E_i can be described below,

$$P_i = \{p_i = (x_i, y_i) | x_i = a_i\cos\theta\cos\phi_i - b_i\sin\theta\sin\phi_i + x_{i0},$$
$$y_i = a_i\cos\theta\sin\phi_i + b_i\sin\theta\cos\phi_i + y_{i0}\}, \theta \in [0, 2\pi], \quad (2.2)$$

where x_i and y_i denote the coordinate of the point on E_i, and θ is the corresponding angle for each point.

2.2.2 Minimum distance searching algorithm

This section presents the algorithm that calculates the minimum distance between two ellipses in a searching method. One objective of this chapter is to find the optimal path such that a group of agents achieves a predefined formation. The obstacle avoidance must be considered because of the shape of each agent. Thus, it is necessary to calculate the minimum distances among agents and the corresponding points. Also, the algorithm of calculating tangents between elliptical agents, which will be given in next subsection, need to be proposed to deal with the collision problem. The minimum distance between two ellipses is given in Figure 2.1.

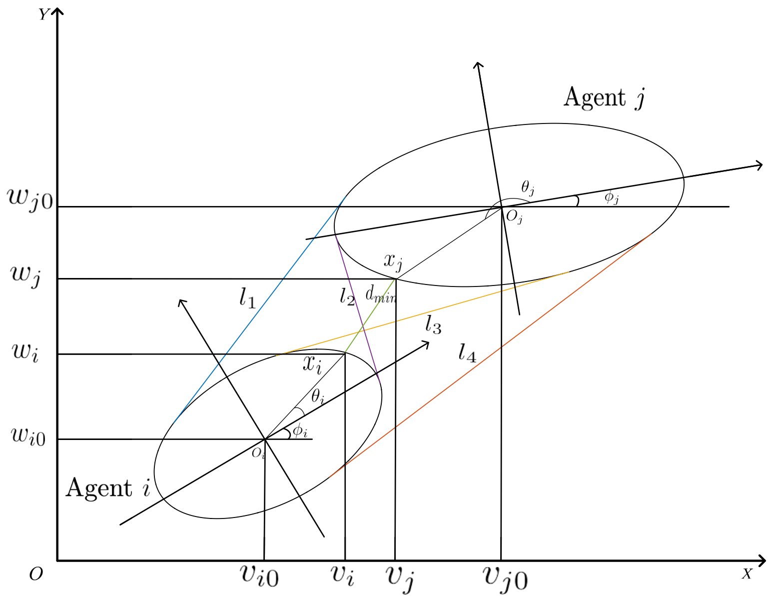

FIGURE 2.1: Minimum distance and tangents between two ellipses

The minimum distance searching algorithm is given below. Firstly, two elliptical agent E_i and E_j centered at (x_{i0}, y_{i0}) and (x_{j0}, y_{j0}) with heading angle ϕ_i and ϕ_j are defined respectively. The long axis and short axis are given as a_i, a_j and b_i, b_j, respectively. Randomly pick two points p_i and p_j as initial points on E_i and E_j. The corresponding angles can be written as θ_i and θ_j. To find a minimum distance between E_i and E_j, parameter δ is employed to change the angle θ_i and θ_j to find the optimal points for the minimum distance. The angle set can be expressed as

$$\begin{aligned} \Theta_i &= \{\theta_i - \delta, \theta_i, \theta_i + \delta\}, \\ \Theta_j &= \{\theta_j - \delta, \theta_j, \theta_j + \delta\}. \end{aligned} \tag{2.3}$$

The minimum distance between two elliptical agents can be obtained by comparison among distances of the points based on the angle set. In (2.3), each angle set maps three points on one elliptical agent. Then the distances between each point on E_i and other three points on E_j are calculated. The distance set in kth term can be expressed as follows,

$$\begin{aligned} D^k &= \{d_1^k, d_2^k, \ldots, d_9^k\} \\ &= \{|p_i - p_j|, \theta_i \in \Theta_i, \theta_j \in \Theta_j\}, \end{aligned}$$

and the minimum distance in the kth term can be expressed as

$$\begin{aligned} d_{min}^k &= \min D^k \\ &= \min_{\theta_i \in \Theta_i, \theta_j \in \Theta_j} |p_i - p_j|. \end{aligned}$$

The values of relative angles can be returned as θ_{imin}^k and θ_{jmin}^k, respectively. Then, compare the relative angles with Θ_i and Θ_j. If

$$\begin{aligned} \theta_{imin}^k &= \theta_i, \\ \theta_{jmin}^k &= \theta_j, \end{aligned}$$

then the minimum distance between E_i and E_j is the distance between point p_i and p_j. Otherwise, θ_i and θ_j will be assigned to θ_{imin}^k and θ_{jmin}^k, respectively, and they will be the initial angles in next iteration. This algorithm will continue to loop until the suitable angles θ_{imin} and θ_{jmin} are found. The minimum distance between these two ellipses can be expressed as

$$d_{min} = |p_i - p_j|,$$

where p_i and p_j are obtained in (2.2), while $\theta_i = \theta_{imin}$ and $\theta_j = \theta_{jmin}$.

The minimum distance searching algorithm is given follows.

Algorithm 1 Minimum distance searching algorithm

Input:

The coordinate of the center of agent E_i, $p_i^c = (x_{i0}, y_{i0})$;

The coordinate of the center of agent E_j, $p_j^c = (x_{j0}, y_{j0})$;

The long axis and short axis of agent E_i, a_i and b_i;

The long axis and short axis of agent E_j, a_j and b_j;

The heading angle of agent E_i, ϕ_i;

The heading angle of agent E_j, ϕ_j;

Output:

Find the minimum distance between agent E_i and E_j, d_{min};

1: $[d_{min}] = mindis(a_i, b_i, a_j, b_j, p_i^c, p_j^c, \phi_i, \phi_j)$;
2: Randomly generate starting angle θ_i, $\theta_i = 2 \times \pi \times rand(1)$;
3: Randomly generate starting angle θ_j,$\theta_j = 2 \times \pi \times rand(1)$,
4: $\delta = \frac{1.8}{\pi}$;
5: Calculate the initial angle set Θ_i, $\Theta_i = \{\theta_i - \delta, \theta_i, \theta_i + \delta\}$;
6: Calculate the initial angle set Θ_j, $\Theta_j = \{\theta_j - \delta, \theta_j, \theta_j + \delta\}$;
7: Based on the angle in angle set Θ_i, the corresponding point on E_i can be calculated by

for n=1:3 **do**
    $x_1(n) = a_i * cos(\Theta_i(n)) * cos(\phi_i) - b_i * sin(\Theta_i(n)) * sin(\phi_i) + x_{i0}$;
    $y_1(n) = a_i * cos(\Theta_i(n)) * sin(\phi_i) + b_i * sin(\Theta_i(n)) * cos(\phi_i) + y_{i0}$;
end for
Based on the angle in angle set Θ_j, the corresponding point on E_j can be calculated by

for n=1:3 **do**
    $x_2(n) = a_j * cos(\Theta_j(n)) * cos(\phi_j) - b_j * sin(\Theta_j(n)) * sin(\phi_j) + x_{j0}$;
    $y_2(n) = a_j * cos(\Theta_j(n)) * sin(\phi_j) + b_j * sin(\Theta_j(n)) * cos(\phi_j) + y_{j0}$;
end for
for n=1:3 **do**
    for m=1:3 **do**
        $value(n,m) = |\sqrt{(x_1(n) - x_2(m))^2 + (y_1(n) - y_2(m))^2}|$;
    end for
end for
$value_{min} = \min value$;
if $value_{min}$ corresponds to $\Theta_i(2)$ and $\Theta_j(2)$ **then**
    $d_{min} = value_{min}$
else
    θ_i and θ_j are valued as the corresponded angle based on $value_{min}$
    Find the minimum distance d_{min} until $value_{min}$ correspond to $\Theta_i(2)$ and $\Theta_j(2)$;
end if

The flow chart of minimum distance searching algorithm is given in Figure 2.2.

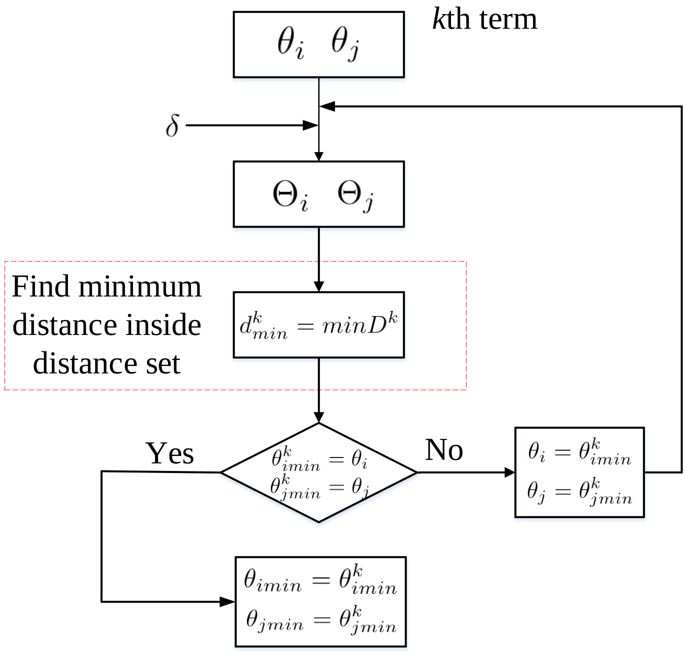

FIGURE 2.2: Flow chart of minimum distance searching algorithm

2.2.3 Tangents searching algorithm

The collision avoidance algorithm in this chapter is based on removing collision angles. Collision angles here mean the angles that are obtained by the tangents from one agent to other agents in the way of its path, which is between the current position of this agent and its objective position. Hence, it is important to find the tangents between two ellipses. Assume that P_i and P_j as the set of all points on agent E_i and agent E_j separately, while p_{itan} and p_{jtan} are corresponded to the specific points both on the ellipses and the tangents between two ellipses. The points p_{itan} and p_{jtan} should satisfy one of the following conditions:
If any $p_i \in P_i$ and $p_j \in P_j$, then

Condition 1 $S = \{(p_{itan} - p_{jtan})V_1(p_i - x_{jtan})^T \cup (p_{itan} - p_{jtan})V_1(p_j - p_{jtan})^T\}$,
for any $s_1, s_2 \in S$, $s_1 \times s_2 \geq 0$

Condition 2 $S = \{(p_{itan} - x_{jtan})V_2(p_i - p_{jtan})^T \cup (p_{itan} - p_{jtan})V_2(p_j - p_{jtan})^T\}$,
for any $s_1, s_2 \in S$, $s_1 \times s_2 \geq 0$

Condition 3 $S = \{(p_{itan} - p_{jtan})V_1(p_i - p_{jtan})^T \cup (p_{itan} - p_{jtan})V_2(p_j - p_{jtan})^T\}$,
for any $s_1, s_2 \in S$, $s_1 \times s_2 \geq 0$

Condition 4 $S = \{(p_{itan} - p_{jtan})V_2(p_i - p_{jtan})^T \cup (p_{itan} - p_{jtan})V_1(p_j - p_{jtan})^T\}$,
for any $s_1, s_2 \in S$, $s_1 \times s_2 \geq 0$

here, $V_1 = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$ and $V_2 = \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix}$.

Similar to the minimum distance searching algorithm, the tangents can also be obtained by the searching algorithm. Parameter δ is employed to vary the initial angles θ_{in} and θ_{jn} in E_i and E_j. The angle set can be defined as $\Theta_i = \{\theta_{in} - \delta,\ \theta_{in},\ \theta_{in} + \delta\}$ and $\Theta_j = \{\theta_{jn} - \delta,\ \theta_{jn},\ \theta_{jn} + \delta\}$. In each iteration, three points on each ellipses will be found based on the angle set. Nine lines between these relative points are defined as $L : \{(p_i - p_j), \theta_i \in \Theta_i, \theta_j \in \Theta_j\}$. The line between unchanged points is called initial line l_{in}. If l_{in} satisfies one of the condition, it can be seen as one of the possible tangents, while $\theta_{ifinal} = \theta_{in}$ and $\theta_{jfinal} = \theta_{jn}$. Otherwise, the points corresponded to the satisfied line would be picked to enter another loop. The algorithm will terminate if all corresponding lines are founded. The tangents searching algorithm under condition 1 is given as follows:

Algorithm 2 Tangents searching algorithm under condition 1

Input:

The coordinate of the center of agent E_i, $p_i^c = (x_{i0}, y_{i0})$;

The coordinate of the center of agent E_j, $p_j^c = (x_{j0}, y_{j0})$;

The long axis and short axis of agent E_i, a_i and b_i;

The long axis and short axis of agent E_j, a_j and b_j;

The heading angle of agent E_i, ϕ_i;

The heading angle of agent E_j, ϕ_j;

Output:

Find the tangent points between agent E_i and E_j under condition 1, $p_{itan} = (x_{itan}, y_{itan})$ and $p_{jtan} = (x_{jtan}, y_{jtan})$;

Randomly generate starting angle θ_{in}, $\theta_{in} = 2 \times \pi \times rand(1)$;
Randomly generate starting angle θ_{jn},$\theta_{jn} = 2 \times \pi \times rand(1)$;
$\delta = \frac{1.8}{\pi}$;
$[p_{itan}, p_{jtan}, \theta_{in}, \theta_{jn}] = tangent(a_i, b_i, a_j, b_j, p_i^c, p_j^c, \phi_i, \phi_j, \theta_{in}, \theta_{jn})$;
Calculate the initial angle set Θ_{in}, $\Theta_{in} = \{\theta_{in} - \delta, \theta_{in}, \theta_{in} + \delta\}$;
Calculate the initial angle set Θ_{jn}, $\Theta_{jn} = \{\theta_{jn} - \delta, \theta_{jn}, \theta_{jn} + \delta\}$;
Based on the angle in angle set Θ_i, the corresponding point on E_i can be calculated by

for n **do**=1:3
    $x_1(n) = a_i * cos(\Theta_{in}(n)) * cos(\phi_i) - b_i * sin(\Theta_{in}(n)) * sin(\phi_i) + x_{i0}$;
    $y_1(n) = a_i * cos(\Theta_{in}(n)) * sin(\phi_i) + b_i * sin(\Theta_{in}(n)) * cos(\phi_i) + y_{i0}$;
end for
Based on the angle in angle set Θ_j, the corresponding point on E_j can be calculated by

for n **do**=1:3
    $x_2(n) = a_j * cos(\Theta_{jn}(n)) * cos(\phi_j) - b_j * sin(\Theta_{jn}(n)) * sin(\phi_j) + x_{j0}$;

    $y_2(n) = a_j * cos(\Theta_{jn}(n)) * sin(\phi_j) + b_j * sin(\Theta_{jn}(n)) * cos(\phi_j) + y_{j0};$
end for
if check01$([x_1(2), y_1(2)], [x_2(2), y_2(2)], [x_1, y_1], [x_2, y_2])$ **then**
    $p_{itan} = (x_1(2), y_1(2));$
    $p_{jtan} = (x_2(2), y_2(2));$
else if check01$([x_1(1), y_1(1)], [x_2(1), y_2(1)], [x_1, y_1], [x_2, y_2])$ **then**
    $\theta_{in} = \Theta_{in}(1);$
    $\theta_{jn} = \Theta_{jn}(1);$
    $[p_{itan}, p_{jtan}, \theta_{in}, \theta_{jn}] = tangent(a_i, b_i, a_j, b_j, p_i^c, p_j^c, \phi_i, \phi_j, \theta_{in}, \theta_{jn});$
else if check01$([x_1(1), y_1(1)], [x_2(2), y_2(2)], [x_1, y_1], [x_2, y_2])$ **then**
    $\theta_{in} = \Theta_{in}(1);$
    $\theta_{jn} = \Theta_{jn}(2);$
    $[p_{itan}, p_{jtan}, \theta_{in}, \theta_{jn}] = tangent(a_i, b_i, a_j, b_j, p_i^c, p_j^c, \phi_i, \phi_j, \theta_{in}, \theta_{jn});$
else if check01$([x_1(1), y_1(1)], [x_2(3), y_2(3)], [x_1, y_1], [x_2, y_2])$ **then**
    $\theta_{in} = \Theta_{in}(1);$
    $\theta_{jn} = \Theta_{jn}(3);$
    $[p_{itan}, p_{jtan}, \theta_{in}, \theta_{jn}] = tangent(a_i, b_i, a_j, b_j, p_i^c, p_j^c, \phi_i, \phi_j, \theta_{in}, \theta_{jn});$
else if check01$([x_1(2), y_1(2)], [x_2(1), y_2(1)], [x_1, y_1], [x_2, y_2])$ **then**
    $\theta_{in} = \Theta_{in}(2);$
    $\theta_{jn} = \Theta_{jn}(1);$
    $[p_{itan}, p_{jtan}, \theta_{in}, \theta_{jn}] = tangent(a_i, b_i, a_j, b_j, p_i^c, p_j^c, \phi_i, \phi_j, \theta_{in}, \theta_{jn});$
else if check01$([x_1(2), y_1(2)], [x_2(3), y_2(3)], [x_1, y_1], [x_2, y_2])$ **then**
    $\theta_{in} = \Theta_{in}(2);$
    $\theta_{jn} = \Theta_{jn}(3);$
    $[p_{itan}, p_{jtan}, \theta_{in}, \theta_{jn}] = tangent(a_i, b_i, a_j, b_j, p_i^c, p_j^c, \phi_i, \phi_j, \theta_{in}, \theta_{jn});$
else if check01$([x_1(3), y_1(3)], [x_2(1), y_2(1)], [x_1, y_1], [x_2, y_2])$ **then**
    $\theta_{in} = \Theta_{in}(3);$
    $\theta_{jn} = \Theta_{jn}(1);$
    $[p_{itan}, p_{jtan}, \theta_{in}, \theta_{jn}] = tangent(a_i, b_i, a_j, b_j, p_i^c, p_j^c, \phi_i, \phi_j, \theta_{in}, \theta_{jn});$
else if check01$([x_1(3), y_1(3)], [x_2(2), y_2(2)], [x_1, y_1], [x_2, y_2])$ **then**
    $\theta_{in} = \Theta_{in}(3);$
    $\theta_{jn} = \Theta_{jn}(2);$
    $[p_{itan}, p_{jtan}, \theta_{in}, \theta_{jn}] = tangent(a_i, b_i, a_j, b_j, p_i^c, p_j^c, \phi_i, \phi_j, \theta_{in}, \theta_{jn});$
else if check01$([x_1(3), y_1(3)], [x_2(3), y_2(3)], [x_1, y_1], [x_2, y_2])$ **then**
    $\theta_{in} = \Theta_{in}(3);$
    $\theta_{jn} = \Theta_{jn}(3);$
    $[p_{itan}, p_{jtan}, \theta_{in}, \theta_{jn}] = tangent(a_i, b_i, a_j, b_j, p_i^c, p_j^c, \phi_i, \phi_j, \theta_{in}, \theta_{jn}).$
end if

The checking algorithm under condition 1 is given in Algorithm 3.

Algorithm 3 Checking condition 1

Input:

The point for checking on E_i, $p_{itan} = (x_{itan}, y_{itan})$;

The point for checking on E_j, $p_{jtan} = (x_{jtan}, y_{jtan})$;

The long axis and short axis of E_i, a_i and b_i;

The long axis and short axis of E_j, a_j and b_j;

The heading angles of E_i and E_j, ϕ_i and ϕ_j;

Output:

Find if the points for checking satisfy condition 1, *true*;

for n=1:3 **do**

$$value_1(n) = (p_{itan} - p_{jtan}) * \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} * ([x_1(n), y_1(n)] - p_{jtan})^T;$$

end for

for n=1:3 **do**

$$value_1(n) = (p_{itan} - p_{jtan}) * \begin{bmatrix} 0 & 1 \\ -1 & 0 \end{bmatrix} * ([x_2(n), y_2(n)] - p_{jtan})^T;$$

end for

$true = (\sum value_1 < 0 == 0)$ and $(\sum value_2 < 0 == 0)$;

The flow chart of left tangent searching algorithm is shown in Figure 2.3, and other three tangents searching methods are similar with tangents searching algorithm of left tangent. Four tangents between two ellipses are given in Figure 2.1.

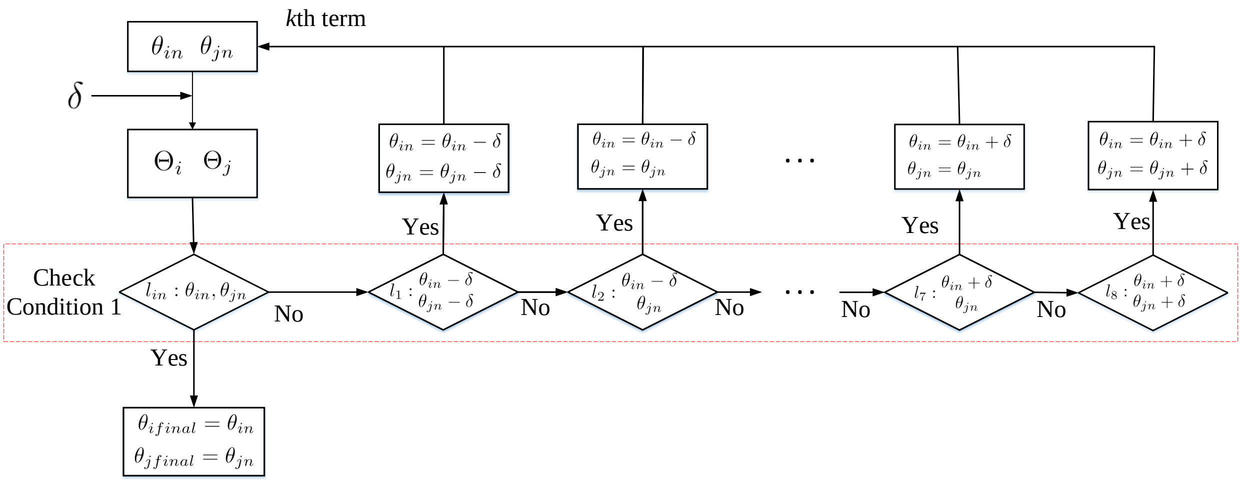

FIGURE 2.3: Flow chart of tangents searching algorithm under Condition 1

2.3 Formation control design

2.3.1 Problem statement

The objective of this chapter is to drive the elliptical agents to the desired formation based on the reference map. Assume that there are N robots in the multi-agent system. The set of the agents is written as

$$E = \{E_1 \quad E_2 \quad E_3 \dots E_N\}. \tag{2.4}$$

The centering points of the individual agents are

$$P^c = \{p_1^c \quad p_2^c \dots p_N^c\}. \tag{2.5}$$

The point set on the edge of the elliptical agents is defined as

$$P = \{P^{E_1} \quad P^{E_2} \quad P^{E_3} \dots P^{E_N}\}. \tag{2.6}$$

Each agent in the group is modeled in first-order dynamics. The centering point of E_i can be represented as p_i^c based on (2.5). Therefore, the dynamic of agent E_i is defined as

$$\begin{bmatrix} \dot{p}_i^c \\ \dot{\phi}_i \end{bmatrix} = \begin{bmatrix} u_i \\ u_{\phi_i} \end{bmatrix}. \tag{2.7}$$

The position control input vector for agent E_i is represented as $u_i = [u_{x_{i0}}, u_{y_{i0}}]$, while $p_i^c = [x_{i0}, y_{i0}]$ denotes the centering position of agent E_i. The heading angle control input for agent E_i is represented as u_{ϕ_i}, while ϕ_i denotes the heading angle of agent E_i. The position control u_i will be designed in Section 2.3.2, while the heading angle control u_{ϕ_i} is obtained by the self-center-base rotation algorithm in Section 2.3.5.

The following assumptions are imposed throughout the chapter to develop our main results in sequel.

Assumption 1. *All agents can move in any 2D directions.*

Note that, by Assumption 1, there is no fixed direction towards which agents are forced to move or rotate. Agent can move along the direction that is planned by the mapping and collision avoidance algorithm without rotation. Rotation control for agent is considered for collision avoidance among agents, and an agent rotates by itself whenever it is too close to another agent. In the process, heading angle ϕ has to be changed to enlarge the minimum distance between two elliptical agents.

Regarding the sensing ability and avoidance range among agents, we introduce the following assumption.

Assumption 2.

For any $p_{sen} \in P_{sen}$, there is a corresponding $p_i \in P_i$ that satisfies

$$|p_i - p_{sen}| \leq R_{sen}.$$

For any $p_{avo} \in P_{avo}$, there a is corresponding $p_i \in P_i$ that satisfies

$$|p_i - p_{avo}| \leq R_{avo}.$$

The center of sensing area and avoidance area of agent E_i is represented by p_i, while p_{sen} and p_{avo} represent the points on sensing ellipse and avoidance ellipse respectively. The points set on sensing ellipse are represented as P_{sen}, while points set on avoidance ellipse are represented as P_{avo}. The radius of sensing range and avoidance range are R_{sen} and R_{avo}, respectively. Notice that Assumption 2 specifies the sensing ability and the avoidance range for each agent. Displacement sensors on agents E_i and E_j have circular ranges, which are centered at p_i and p_j, while p_i and p_j are distributed in the edge of each agent. The sensing area and avoidance area are enclosed by oval surface, as shown in Figure 2.4. Each agent has the same sensing range R_{sen} and avoidance range R_{avo}. Based on E, R_{sen} and R_{avo}, agent could have its own neighbourhood and avoidance group as,

$$\begin{aligned} H_i &= \{H_{E_1}(R_{sen}) \quad H_{E_2}(R_{sen}) \quad \dots \quad \}, \\ H_{E_i}(R_{sen}) &= \{E_1^{H_{E_i}(R_{sen})} \quad E_2^{H_{E_i}(R_{sen})} \quad \dots \quad \}, \end{aligned} \tag{2.8}$$

$$\begin{aligned} A_i &= \{A_{E_1}(R_{avo}) \quad A_{E_2}(R_{avo}) \quad \dots \quad \}, \\ A_{E_i}(R_{avo}) &= \{E_1^{A_{E_i}(R_{avo})} \quad E_2^{A_{E_i}(R_{avo})} \quad \dots \quad \}, \end{aligned} \tag{2.9}$$

where for $p \in P^{E_i}$, $p' \in P^{H_{E_i}(R_{sen})}$, if $|p - p'| \leq R_{sen}$; for $p \in P^{E_i}$, $p' \in P^{A_{E_i}(R_{avo})}$, if $|p - p'| \leq R_{avo}$. In Figure 2.4, for example, the neighbourhood set of E_1 is $H_{E_1}(R_{sen}) = \{E_3, E_4\}$, and its avoidance group is $A_{E_1}(R_{avo}) = \{E_4\}$.

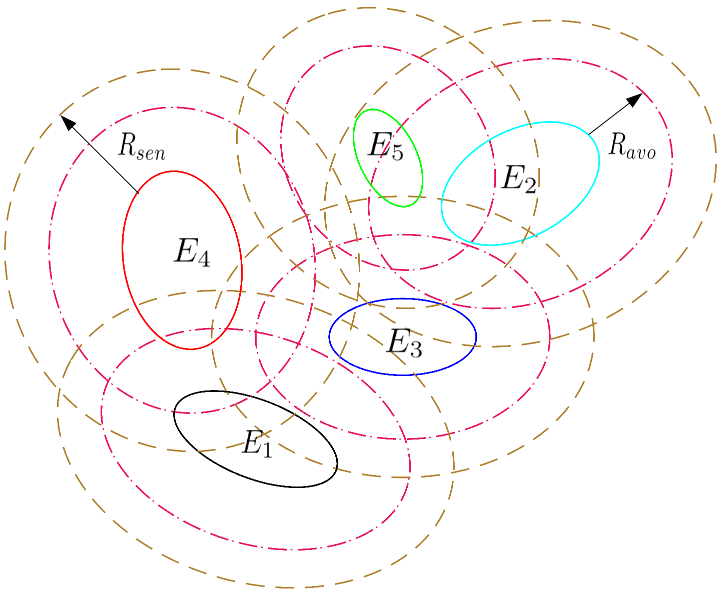

FIGURE 2.4: Formation setup

Assumption 3. *For any $p_i \in P_i$ on agent E_i and any $p_j \in P_j$ on agent E_j, at the initial time t_0, the following condition holds*

$$|p_i - p_j| \;\geq\; \varepsilon,$$

where ε is a positive constant.

It can be seen from Assumption 3 that agents in the group cannot collide with each other in initial state, while ε is big enough to guarantee the initial distances among agents.

We also have the following assumption on the agents.

Assumption 4. *Agents can figure out the center of other agents, at least a portion of which is within their sensing ranges.*

This assumption implies that agents, that cannot be sensed as a whole by agent E_i, can also be considered as the neighbouring agents of E_i and added in $N_{E_i}(R_{sen})$ if at least a portion of them is within the sensing range.
Formation objective. Under Assumptions 1, 2, 3 and 4, for each agent, design a control law u_i, such that all agents can conform to the formation based on the reference map F without collision with the rest of the agents in the group. The reference map F is defined as

$$F = \{f_1 \quad f_2 \quad \dots f_n\}, \tag{2.10}$$

where $f' \in \mathbf{R}^2$ for any $f' \in F$. The set of displacements of f_i is defined as $\bar{F}_{f_i}$ based on F and is given as

$$\bar{F}_{f_i} = \{f_1^{\bar{F}_{f_i}} \quad f_2^{\bar{F}_{f_i}} \quad \dots\},$$

where for any $f \in F$, $f - f_i \in \bar{F}_{f_i}$. This means that agents can achieve the desired formation based on the relative displacements obtained from F, rather than move to some fixed positions in the 2D space. In other words, we design u_i such that

$$\lim_{t\to\infty}(p_i^c - p_j^c) = f_m^{\bar{F}_{f_i}} \neq 0, \tag{2.11}$$

where $p_i^c, p_j^c \in P^c$, $f_m^{\bar{F}_{f_i}} \in \bar{F}_{f_i}$, and $0 \leq t_0 \leq t$.

2.3.2 Controller design

In this section, a controller is proposed to ensure the establishment of the desired formation. Using (2.10), reference map F can be mapped to agents based on the displacements from their neighbourhood. The relative point of agent E_i in F is represented as $f_i \in F$. The controller for each agent is built based on desired formation, excluding the interferences from collision avoidance to another agents. The expectation point g_i in the desired formation for agent E_i should be obtained by its neighbourhood $H_{E_i}(R_{sen})$ and F. Based on $H_{E_i}(R_{sen})$ and f_i, controller on agent E_i generates a momentum u_i under which agent E_i is expected to gradually reach its desired position g_i. For any $E_i \in E$, the agent formation will become stable when $u_i = 0$. The control of each agent in the group is discussed in two situations below.

The controller of an agent with one neighbour is shown in Figure 2.5. It is clear that $H_{E_i}(R_{sen}) = \{E_j\}$, and O_{ij} is the midpoint between E_i and E_j centred at $p_i^c, p_j^c \in P^c$, respectively. The relative points mapping in reference formation F are f_i and f_j. The midpoint between them is $O_{f_{ij}}$. The expectation point for E_i in the objective map is g_i. Without considering interferences from outside, we have

$$\begin{aligned}\overrightarrow{(O_{f_{ij}}, O_{ij})} &= \overrightarrow{(f_i, g_i)},\\ \overrightarrow{(O_{f_{ij}}, f_i)} &= \overrightarrow{(O_{ij}, g_i)},\end{aligned} \tag{2.12}$$

where $O_{ij} = \frac{p_i^c + p_j^c}{2}$, $O_{f_{ij}} = \frac{f_i + f_j}{2}$, which means that the length and center point constituted by f_i and f_j is not changed in desired formation. According to (11), controller on agent E_i generates a momentum under which agent E_i is expected to gradually reach its desired position g_i in the objective map. The coordinate of $g_i = (x_{g_i}, y_{g_i})$ can be found from

$$\begin{aligned} g_i &= f_i + \overrightarrow{(O_{f_{ij}}, O_{ij})} \\ &= f_i + (O_{ij} - O_{f_{ij}}). \end{aligned} \tag{2.13}$$

The control momentum for E_i takes the form as

$$u_i = \overrightarrow{(p_i^c, g_i)} = g_i - p_i^c. \tag{2.14}$$

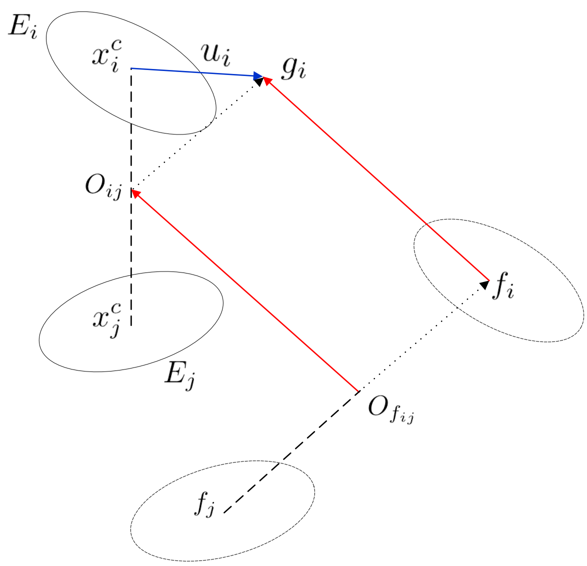

FIGURE 2.5: Agent with one neighbour

The second situation is when an agent has two or more neighbours. The agent and its neighbours form an area instead of a line. In Figure 2.6, it can be seen that the neighbourhood of E_i is $H_{E_i}(R_{sen}) = \{E_j, E_k\}$, which are centered at p_i^c, p_j^c, $p_k^c \in P^c$. The relative points in reference map F are represented by f_i,f_j and f_k, respectively. Centers of these agents form a triangle area, which has a center $O_{ijk} = \frac{x_i^c + x_j^c + x_k^c}{3}$. Correspondingly, the center of f_i, f_j and f_k is written as $O_{f_{ijk}} = \frac{f_i + f_j + f_k}{3}$, while O_{ijk}, $O_{f_{ijk}}$, p_i^c and g_i satisfy

$$\begin{aligned} \overrightarrow{(O_{f_{ijk}}, O_{ijk})} &= \overrightarrow{(f_i, g_i)}, \\ \overrightarrow{(O_{f_{ijk}}, f_i)} &= \overrightarrow{(O_{ijk}, g_i)}. \end{aligned} \tag{2.15}$$

The coordinate of $g_i = (x_{g_i}, y_{g_i})$ is given as

$$\begin{aligned} g_i &= f_i + \overrightarrow{(O_{f_{ijk}}, O_{ijk})} \\ &= f_i + (O_{ijk} - O_{f_{ijk}}). \end{aligned} \tag{2.16}$$

The control momentum for E_i is given as

$$u_i = \overrightarrow{(p_i^c, g_i)} = g_i - p_i^c. \tag{2.17}$$

Hence, based on (2.13) and (2.16), the desired position g_i for agent E_i is obtained by

$$\begin{aligned} g_i &= f_i + \overrightarrow{(O_{f_i}, O_i)} \\ &= f_i + (O_i - O_{f_i}). \end{aligned} \tag{2.18}$$

The general control momentum for agent E_i is proposed as

$$u_i = \overrightarrow{(p_i^c, g_i)} = g_i - p_i^c, \tag{2.19}$$

where O_{f_i} and O_i are derived from $H_{E_i}(R_{sen})$ and F.

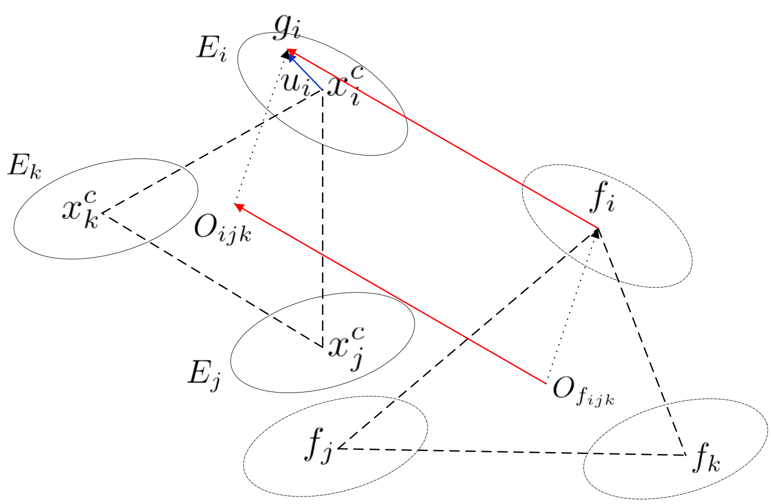

FIGURE 2.6: Controller for agent with two or more neighbours

It can be seen in (2.18) that target position for each agent is obtained from the relative position in reference formation and midpoint of the area constituted by the agent and its neighbours. Meanwhile, the controller for each agent is designed based on target position and its own location.

2.3.3 Random mapping algorithm

To find the optimal mapping for each agent, a random mapping algorithm is built to optimize mapping decisions for the agents. The relative position

f_i for agent E_i can be found from the random mapping algorithm to obtain desired position g_i of E_i. The random mapping algorithm will be given in following steps. In the kth iteration of the operation of whole algorithm, η mapping decisions are generated randomly, and can be defined as

$$\begin{aligned} R^k &= \{r_1^k \quad r_2^k \quad \dots \quad r_\eta^k\}, \\ r_s^k &= \{r_s^k(E_1) \quad r_s^k(E_2) \quad \dots \quad r_s^k(E_n)\}, 1 \leq s \leq \eta. \end{aligned} \tag{2.20}$$

Here, R^k is the set of the generated mapping decisions, in which r_s^k is sth mapping decision. The element in r_s^k corresponds to agents in the reference map. The mapping pool in the kth iteration is

$$M^k = \{R^k \quad r_{op}^{k-1}\}, \tag{2.21}$$

where r_{op}^{k-1} is the optimal mapping decision in the $(k-1)$th iteration. The spacings between agents and mapping positions are calculated based on r_s^k. The sum of corresponding distances set L^k in the kth term can be written as

$$\begin{aligned} L^k &= \{L_1^k \quad L_2^k \quad \dots \quad L_{\eta+1}^k\}, \\ L_s^k &= \sum_{1 \leq i \leq N} |p_i^c - g_s^k(E_i)|, 1 \leq s \leq \eta + 1. \end{aligned} \tag{2.22}$$

In (2.22), L_s^k is the sum of distances between agents' current positions and their desired positions based on r_s^k. The desired position for agent E_i in sth mapping decision is represented by $g_s^k(E_i)$ that can be calculated by (2.18). Based on (2.5) and (2.22), the sum of distances of the agents is obtained and compared. Optimal mapping decision r_{op}^k is the mapping decision corresponding to minimum component in L^k. This mapping algorithm will be terminated when desired formation establishes. Algorithm 4 illustrates the random mapping algorithm.

Algorithm 4 Random mapping algorithm in the kth iteration

Input:

The current positions of the group of elliptical agents in the $k-1$th iteration, $P^c = \{p_1^c \, p_2^c \cdots p_n^c\}$;

The optimal mapping in the $k-1$th iteration, r_{op}^{k-1};

Output:

Find the optimal mapping in the kth iteration, r_{op}^k;

1: Generate η mappings randomly, $R^k = \{r_1^k \, r_2^k \cdots r_\eta^k\}$;
2: Set up the mapping set $M^k = \{R^k \, r_{op}^{k-1}\}$;
3: Calculate the corresponding desired positions based on M^k, the desired position of agent E_i $g_s^k(E_i)$ based on the sth mapping in M^k;
4: Calculate the sum of corresponding distances set in the kth based on the mapping in M^k;
5: $L_s^k = \sum_{1 \leqslant i \leqslant N} |p_i^c - g_s^k(E_i)|, 1 \leqslant s \leqslant \eta + 1$;

6: Find the minimum value from $L^k = \{L_1^k \, L_2^k \cdots L_{\eta+1}^k\}$;
7: The optimal mapping in the kth iteration r_{op}^k is obtained based on the minimum value of L^k.

The flow chart of the mapping algorithm is shown in Figure 2.7.

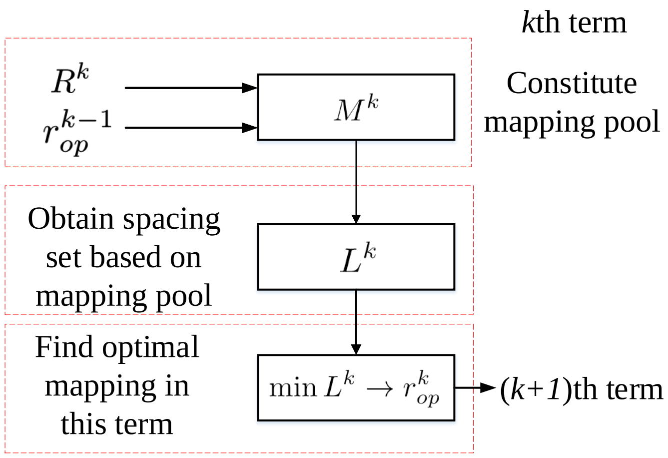

FIGURE 2.7: Random mapping algorithm

Remark 1. *It should be emphasized that the more random mappings are selected in each updated iteration, the less iterations is required to find the global optimal mapping decision. Here, global optimal mapping decision represents the optimal mapping decision of all possible mappings. Assume there are n agents in the system. The number of possible mapping is given as* $n!$. *The probability of the optimal mapping decision to be found by one selection is* $\frac{1}{n!}$. *When* η *random mappings are selected in one term, the probability of choosing the optimal mapping decision becomes* $\eta \times \frac{1}{n!}$. *When* η *increases, the convergence rate to achieve the global optimal mapping decision becomes larger.*

2.3.4 Collision avoidance algorithm

The collision avoidance is used to ensure that agents can move to the desired position without any collision, causing by the sizes of agents. In this section, a collision avoidance algorithm for elliptical agents is proposed. The algorithm is based on the avoidance group of each agent and avoidance angles. Here, avoidance group of an agent is the agent set of other agents in the group within its avoidance range. Collision angles represent the angles between two insect tangents of agent and agents within its avoidance group. Tangents among each agent can be obtained based on the tangents searching algorithm given in Section 2.3.

For an agent $E_i \in E$, its avoidance range $A_{E_i}(R_{avo})$ is defined in (2.9). Let $E_j \in A_{E_i}(R_{avo})$. The intersect tangents between E_i and E_j are l_2^{ij} and l_3^{ij}, respectively. Tangent points on each agent are denoted by $p_i^{l_2^{ij}}$, $p_i^{l_3^{ij}}$, $p_j^{l_2^{ij}}$ and $p_j^{l_3^{ij}}$, respectively. Then, the vectors of these two tangents are defined as $\overrightarrow{(p_i^{l_2^{ij}}, p_j^{l_2^{ij}})}$ and $\overrightarrow{(p_i^{l_3^{ij}}, p_j^{l_3^{ij}})}$. It follows that the collision angle between these two tangents is

$$\begin{aligned}\psi_{ij} &= < \overrightarrow{(p_i^{l_2^{ij}}, p_j^{l_2^{ij}})}, \overrightarrow{(p_i^{l_3^{ij}}, p_j^{l_3^{ij}})} > \\ &= \arccos(\frac{(p_j^{l_2^{ij}} - p_i^{l_2^{ij}}) \cdot (p_j^{l_3^{ij}} - p_i^{l_3^{ij}})}{|p_j^{l_2^{ij}} - p_i^{l_2^{ij}}||p_j^{l_3^{ij}} - p_i^{l_3^{ij}}|}). \end{aligned} \tag{2.23}$$

The avoidance angle set for E_i is obtained as

$$\Psi_i = \{\psi_{ij}, E_j \in A_{E_i}(R_{avo})\}. \tag{2.24}$$

Then the angle of initial control monument u_i obtained in (2.19) should be checked with Ψ_i. First, for any $E_j \in A_{E_i}(R_{avo})$, midline between two intersect tangents need to be calculated as follow,

$$m_{ij} = \frac{\overrightarrow{(p_i^{l_2^{ij}}, p_j^{l_2^{ij}})}}{|(p_i^{l_2^{ij}}, p_j^{l_2^{ij}})|} + \frac{\overrightarrow{(p_i^{l_3^{ij}}, p_j^{l_3^{ij}})}}{|(p_i^{l_3^{ij}}, p_j^{l_3^{ij}})|}. \tag{2.25}$$

The set of midline of E_i based on $A_{E_i}(R_{avo})$ and Ψ_i is written as

$$M_i = \{m_{ij}, E_j \in A_{E_i}(R_{avo})\}. \tag{2.26}$$

The angles between u_i and $m_{ij} \in M_i$ can be worked out as

$$\begin{aligned}\varphi_{ij} &= < u_i, m_{ij} > \\ &= \arccos(\frac{u_i \cdot m_{ij}}{|u_i||m_{ij}|}). \end{aligned} \tag{2.27}$$

The set of φ_{ij} is given as

$$\Omega_i = \{\varphi_{ij}, E_j \in A_{E_i}(R_{avo})\}. \tag{2.28}$$

To correct the control momentum u_i based on collision avoidance, the comparison between $\frac{\psi_{ij}}{2}$ and $\varphi_{ij} \in \Omega_i$ has to be given. If all $\varphi_{ij} \geq \frac{\psi_{ij}}{2}$, $\varphi_{ij} \in \Omega_i$, $\psi_{ij} \in \Psi_i$, this means the moving direction of E_i is out of the possible avoidance areas which are constituted by intersect tangents, and the agent can move based on u_i. The desired position g_i in (2.18) will not be changed. If there is $\varphi_{ij} < \frac{\psi_{ij}}{2}$, $\varphi_{ij} \in \Omega_i$, $\psi_{ij} \in \Psi_i$, then control monument u_i will move through possible avoidance area. In this situation, to avoid possible collision between agents, center O_i in (2.18) has to be changed. The desired position for E_i has to be changed based on O_i, and a modified control algorithm should be given. The changing g_i of agent E_i is represented by g_i^d, and the modified control monument is represented by u_i^d. To find u_i^d, the minimum angle between u_i and tangents of E_i is given as

$$\vartheta_i = \min < u_i, \overrightarrow{(p_i^{l_k^{ij}}, p_j^{l_k^{ij}})} >, \tag{2.29}$$

where $k = 2, 3$, and E_j satisfies $\varphi_{ij} < \frac{\psi_{ij}}{2}$. The corresponding angle of control momentum for E_i is changed to the angle of vector $\overrightarrow{(p_i^{l_k^{il}}, p_l^{l_k^{il}})}$, which corresponds to ϑ_i. Here, $p_l^{l_k^{il}}$ is with respect to the agent E_l based on ϑ_i. The value of k corresponds to ϑ_i. The length of control vector u_i^d is given as

$$|u_i^d| = \xi(d_{il}^{min} - d_{stop}), \tag{2.30}$$

where d_{il}^{min} is the minimum distance between E_i and E_l, which can be obtained by minimum distance searching algorithm in Section 2.2, while ξ is a positive coefficient, and $0 < \xi \leq 1$. A default distance d_{stop} is given in order to prevent collision further. Agents will stop moving when their minimum distance $d_{min}^{il} \leq d_{stop}$. Based on (2.19), (2.29) and (2.30), the developed control monument can be formed as

$$u_i^d = |u_i^d| \frac{(p_i^{l_k^{il}}, p_l^{l_k^{il}})}{|(p_i^{l_k^{il}}, p_l^{l_k^{il}})|}, \tag{2.31}$$

while agent E_l and k are corresponding to ϑ_i. The changed desired position g_i^d for E_i can be formed as

$$g_i^d = p_i^c + u_i^d. \tag{2.32}$$

Remark 2. *According to (2.30), ξ is relevant to the length of u_i^d, which leads to the moving distance that an agent intends to go. Agent E_l is the nearest to E_i in the direction u_i, and E_l also travels among $\overrightarrow{(p_l^{l_k^{il}}, p_i^{l_k^{il}})}$. To avoid collision between agents E_i and E_l, the distance d_{il} should be ensured to be smaller than d_{il}^{min}, while d_{il} is the sum of the distance, which E_i and E_l travel, projected onto the direction of d_{il}^{min}. In (2.30), ξ and d_{stop} are used to avoid collision among agents, while $d_{min}^{il} - d_{stop}$ is defined as the safety distance between agents E_i and E_j. It can be seen from (2.30) that*

$$d_{il} = |u_i^d| \times \alpha_i + |u_l^d| \times \alpha_l, \tag{2.33}$$

where

$$\alpha_i =< u_i^d, d_{il}^{min} >= \arccos(\frac{u_i^d \cdot d_{il}^{min}}{|u_i^d||d_{il}^{min}|}),$$
$$\alpha_l =< u_l^d, d_{li}^{min} >= \arccos(\frac{u_l^d \cdot d_{li}^{min}}{|u_l^d||d_{li}^{min}|}).$$

It is clear that, $|d_{il}^{min}| = |d_{li}^{min}|$. *To avoid the collision between agent* E_i *and* E_l, *the travel distance* d_{il} *should be satisfied*

$$d_{il} < d_{il}^{min} - d_{stop}, \tag{2.34}$$

which is

$$\xi(d_{il}^{min} - d_{stop}) \times \alpha_i + \xi(d_{il}^{min} - d_{stop}) \times \alpha_l < d_{il}^{min} - d_{stop}. \tag{2.35}$$

It can be seen that when ξ *satisfies*

$$0 < \xi < \min\{\frac{0.5}{\cos\alpha_i}, \frac{0.5}{\cos\alpha_l}\}, \tag{2.36}$$

there will be no collision between agents E_i *and* E_l. *The value of angle* ψ_{il} *between two insect tangents between* E_i *and* E_l *satisfies* $\psi_{il} \in (0, \pi)$, *and* d_{il}^{min} *crosses by the area between two insect tangents, as seen in Figure 2.2, thus* α_i *satisfies* $\alpha_i \in (0, \frac{\pi}{2})$, *and* α_l *satisfies* $\alpha_l \in (0, \frac{\pi}{2})$. *Substituting* $\alpha_i \in (0, \frac{\pi}{2})$ *and* $\alpha_l \in (0, \frac{\pi}{2})$ *into (2.36), we have that when* $0 < \xi \leq 0.5$, *agents* E_i *and* E_l *will never collide each other.*

In this section, the collision avoidance algorithm is developed to guarantee the agents can go to the desired formation without any collisions. The key of this algorithm is to find an optimal path by removing the possible collision areas on the path of each agent based on collision angles. The collision avoidance algorithm is given in Algorithm 5.

Algorithm 5 The collision avoidance algorithm

Input:

The intersection tangents information from tangent searching algorithm, l_2^{ij} and l_3^{ij}, while $E_i \in E, E_j \in E$;

The avoidance range of each agent, $A_{E_i}(R_{avo}), E_i \in E$;

The minimum distance set between each agent, d_{ij}^{min}, $E_i \in E, E_j \in E$;

The original control input of each agent, $u_i, E_i \in E$;

The constant parameter, d_{stop};

Output:

The updated control law of the elliptical agents, $u_i^d, E_i \in E$;

for i=1:N **do**
  Calculate the angle between two intersection ψ_{ij}, where $E_j \in A_{E_N}(R_{avo})$;
  Set up the angle set of ψ_{ij}, Ψ_i;
  Calculate the midline m_{ij} between l_2^{ij} and l_3^{ij};
  Calculate the angle φ_{ij} between m_{ij} and u_i;
  Set up the angle set of φ_{ij}, Ω_i;
  Compare the elements in Ψ_n and Ω_i;
  if all $\varphi_{ij} \in \Omega_i \geq \frac{\psi_{ij}}{2}$ **then**
    $u_i^d = u_i$;
  else

```
        Find the minimum angle between u_i and the tangents of E_N;
        ϑ_n = min < u_i, (p_i^{l_k^{il}}, p_l^{l_k^{il}}) >, where E_l ∈ A_{E_i}(R_avo), and k = 1,2;
        |u_i^d| = ξ(d_il^min − d_stop);
        u_i^d = |u_i^d| (p_i^{l_k^{il}}, p_l^{l_k^{il}}) / |(p_i^{l_k^{il}}, p_l^{l_k^{il}})|;
    end if
end for
```

In these lines: $\vartheta_n = \min < u_i, \overrightarrow{(p_i^{l_k^{il}}, p_l^{l_k^{il}})} >$, where $E_l \in A_{E_i}(R_{avo})$, and $k = 1, 2$; $|u_i^d| = \xi(d_{il}^{min} - d_{stop})$; $u_i^d = |u_i^d| \frac{(p_i^{l_k^{il}}, p_l^{l_k^{il}})}{|(p_i^{l_k^{il}}, p_l^{l_k^{il}})|}$;

The flow chart of collision avoidance algorithm is shown in Figure 2.8.

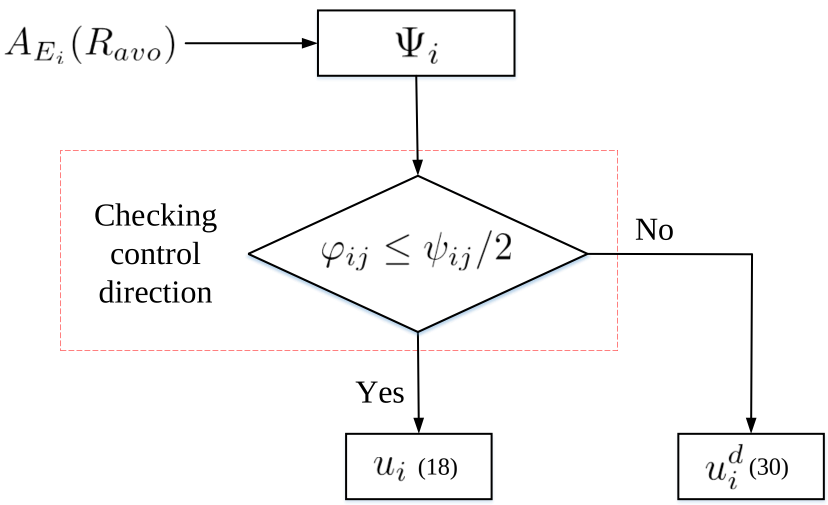

FIGURE 2.8: Collision avoidance algorithm

2.3.5 Self-center-based rotation algorithm

In order to further avoid collision among agents, an algorithm of self-center-based rotation is proposed. It is a direct way to increase the minimum distance among agents to reduce the probability of collision. The algorithm can be summarized as follows. For agent E_i, the heading angle ϕ_i is changed based on the agent E_j, which is closest to E_i. The self-center-based rotation algorithm is given as follows:

(i) The heading angle of agents E_i and E_j are represented as ϕ_i and ϕ_j. Let $\phi_i' = \phi_i + \gamma$, $\phi_i'' = \phi_i - \gamma$. γ here is a positive coefficient, and we take a value of $\gamma = 1^o$;

(ii) Calculate the minimum distance based on corresponding heading angles, where d_{ij}^{min} represents the minimum distance between agents with ϕ_i and ϕ_j; $d_{ij}'^{min}$ represents the minimum distance between agents with ϕ_i' and ϕ_j; And $d_{ij}''^{min}$ represents the minimum distance between agents with ϕ_i'' and ϕ_j based on Section 2.2; and

(iii) Compare d_{ij}^{min}, $d_{ij}'^{min}$ and $d_{ij}''^{min}$. Let ϕ_i, which is corresponding to the maximum distance, be the new direction of the elliptical agent.

The flow chart of the self-rotation algorithm is shown in Figure 2.9.

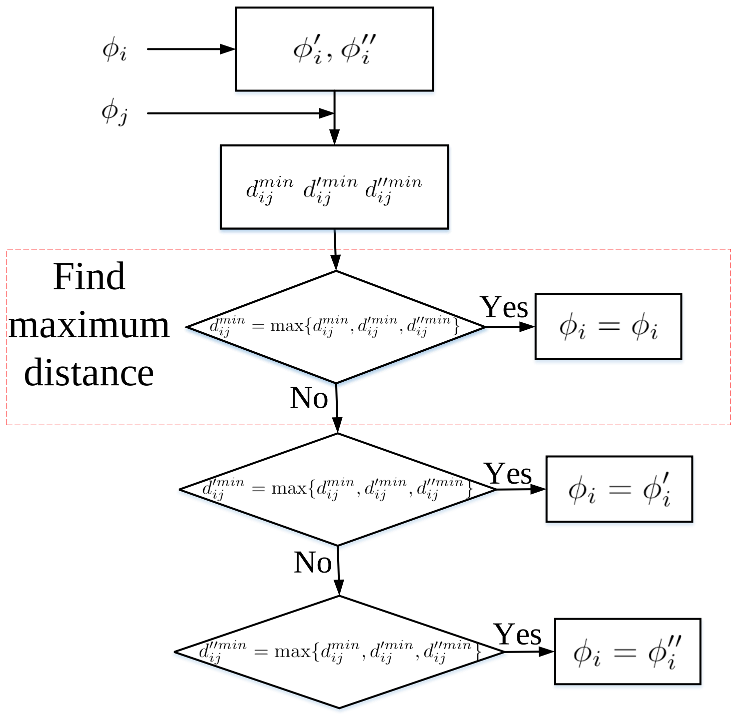

FIGURE 2.9: Self-rotation algorithm

2.3.6 Formation control for multiple elliptical agents without collision

The formation control strategy combined with collision avoidance algorithm for N elliptical agents is proposed in this section. The whole formation control strategy is given in Algorithm 6.

Algorithm 6 Formation control strategy with collision avoidance

Input:

The initial positions of the whole group, P^c;

The long axis set and short axis set of each agent, $AA = \{a_1\, a_2 \cdots a_N\}$ and $BB = \{b_1\, b_2 \cdots b_N\}$;

The heading angle of each agent, $\phi = \{\phi_1\, \phi_2 \cdots \phi_n\}$;

The predefined formation, $F = \{f_1\, f_2 \cdots f_N\}$;

The initial optimal mapping, r_{op}^0;

The sensing range of each elliptical agent, $R_{sen} = \{R_{sen}^1\, R_{sen}^2 \cdots R_{sen}^N\}$;

The avoidance range of each elliptical agent, $R_{sen} = \{R^1_{avo}\ R^2_{avo} \cdots R^N_{avo}\}$;
The constant parameter, $d_s top$;

Output:

Achieve the desired formation which has the same displacements with F.

for t=1:100000000 **do**
  Calculate the minimum distance among the elliptical agents;
  for i=1:N **do**
    for j=1:N **do**
      $[D^t(i,j)] = mindis(AA(i), BB(i), AA(j), BB(j), P^c(i,:), P^c(j,:), \phi(i), \phi(j))$;
    end for
  end for
  Find the optimal mapping in tth term;
  $[r^t_{op}] = Randommapping(P^c, r^{t-1}_{op})$;
  Calculate the controller of each elliptical agent;
  Using R_{sen} find the neighboring group of each agent, $H = \{H_1\ H_2 \cdots H_N\}$;
  for i=1:N **do**
    The desired position of E_i is obtained by $g^t_i = F(r^t_{op}(i)) + \frac{p^c_i + H_i}{length(H_i)} - \frac{F(r^t_{op}(i)) + F(r^t_{op}(H_i))}{length(H_i)}$;
    The controller u^t_i of E_i is calculated;
    $u^t_i = g^t_i - p^c_i$;
  end for
  Start the collision avoidance algorithm;
  for i=1:n **do**
    Using R_{avo} find the neighboring group of each agent, $A = \{A_1\ A_2 \cdots A_N\}$;
    Calculate the tangents of E_i;
    $[l^i_1, l^i_2] = tangent(p_{itan}, p_{jtan}, AA(i), BB(i), \phi_i, AA(j), BB(j), \phi_j))$;
    $[u^d_i] = avo(A(i), l^i_1, l^i_2, u_i, d_{stop}, d^{min}_{ij})$;
    $p^c_i = p^c_i + u^d_i$;
  end for
  Using self-center-based rotation algorithm to make sure the minimum distance among the elliptical agents;
  if Displacements among the elliptical agents in their current positions $-$ Displacements among the elliptical agents in F=0 **then**
    break;
  end if
end for

Though the formation control strategy with collision avoidance algorithm, we can drive the group of agents to our desired formation in any space based on the predefined formation.

2.4 Simulation examples

In this section, simulation results are given to illustrate the effectiveness of the proposed algorithms. It is assumed that five elliptical agents are equipped with displacement sensors of the same limited sensing range and dynamics. These five agents have different shape properties. The sensing range is $R_{sen} = 9$, and the avoidance range is $R_{avo} = 5$. Neighbourhood and avoidance group of each agent can obtained based on R_{sen} and R_{avo}. Parameter δ in minimum distance searching algorithm and tangents searching algorithm is given as 0.01. The team operates with limited communication ability, and agents can only send identities and mapping decision to each other. The coefficients in (2.30) are given as $\xi = 0.3$, and $d_{stop} = 0.1$. In the first two examples, five elliptical agents are employed, and the properties of these agents are given as,

$$\begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = \begin{bmatrix} 5 \\ 3 \end{bmatrix}, \begin{bmatrix} a_2 \\ b_2 \end{bmatrix} = \begin{bmatrix} 6 \\ 4 \end{bmatrix}, \begin{bmatrix} a_3 \\ b_3 \end{bmatrix} = \begin{bmatrix} 4 \\ 2 \end{bmatrix}, \begin{bmatrix} a_4 \\ b_4 \end{bmatrix} = \begin{bmatrix} 6 \\ 3 \end{bmatrix}, \begin{bmatrix} a_5 \\ b_5 \end{bmatrix} = \begin{bmatrix} 7 \\ 4 \end{bmatrix}.$$

The initial positions and heading angles for agents are

$$\begin{bmatrix} x_{10} \\ y_{10} \\ \phi_1 \end{bmatrix} = \begin{bmatrix} 3 \\ -2 \\ 0^\circ \end{bmatrix}, \begin{bmatrix} x_{20} \\ y_{20} \\ \phi_2 \end{bmatrix} = \begin{bmatrix} 11 \\ 10 \\ 37^\circ \end{bmatrix}, \begin{bmatrix} x_{30} \\ y_{30} \\ \phi_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 12 \\ 111^\circ \end{bmatrix}, \begin{bmatrix} x_{40} \\ y_{40} \\ \phi_4 \end{bmatrix} = \begin{bmatrix} -7 \\ -10 \\ 154^\circ \end{bmatrix}, \begin{bmatrix} x_{50} \\ y_{50} \\ \phi_5 \end{bmatrix} = \begin{bmatrix} -12 \\ 4 \\ 97^\circ \end{bmatrix}.$$

The objective map is $F = \{f_1 \quad f_2 \quad \ldots f_N\}$, where

$$f_1 = \begin{bmatrix} -10 \\ 8 \end{bmatrix}, f_2 = \begin{bmatrix} 0 \\ 8 \end{bmatrix}, f_3 = \begin{bmatrix} 10 \\ 8 \end{bmatrix}, f_4 = \begin{bmatrix} -6 \\ -4 \end{bmatrix}, f_5 = \begin{bmatrix} 6 \\ -4 \end{bmatrix}.$$

The set up of the first two simulation examples is: i) to illustrate collision avoidance algorithm and self-rotation algorithm, a fixed displacement is employed; and ii) to further enhance the effectiveness of control scheme in this article, random mapping algorithm is developed.

Example 1 A fixed mapping decision is illustrated in this example. The mapping decision is given as $F_{opt} = \{3 \quad 1 \quad 2 \quad 5 \quad 4\}$, in which elements represent the identities in F. Figure 2.10 shows the trajectories of each elliptical agent in the group based on F_{opt}. For the legend $*$ used in Figure 2.10, it denotes a number that is the identity of an elliptical agent. The legend $\triangle$ used in Figure 2.10, denotes the initial position of each agent. Changes in minimum distance and heading angle of each agent in the group are shown in Figures 2.11 and 2.12, respectively. The misalignment between the temporary formation and desired formation is displayed in Figure 2.13. The curve of the distance between temporary formation and desired formation approaches to 0, uniformly, even though the distance increases in some moments due to the collision avoidance. It can be seen from Figures 2.10-2.13 that there is no collision between any two agents. Moreover, all the agents are driven to the desired formation based on reference map F, even though they have to take some extra effort to avoid collision. The control signals of each agent are shown in Figure 2.14 and Figure 2.15.

Example 2 In this example, random mapping algorithm is used to find optimal mapping decision for each agent. In this algorithm, an initial mapping decision is proposed first. This mapping decision is given as $F_{opt} = \{3 \quad 1 \quad 2 \quad 5 \quad 4\}$. It is assumed that 5 mapping decisions are generated per

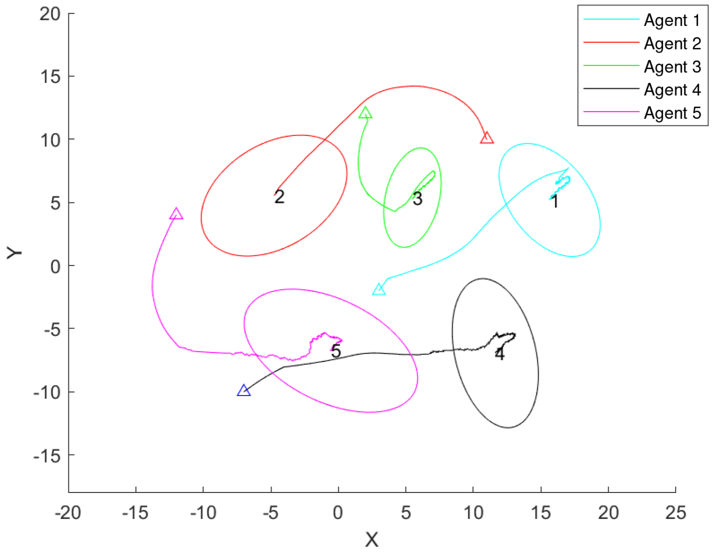

FIGURE 2.10: Trajectories of five elliptical agents

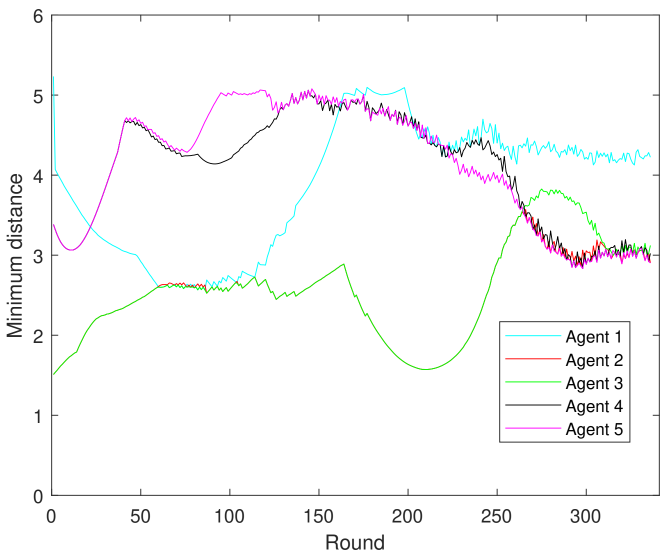

FIGURE 2.11: Changes in minimum distance of each elliptical agent

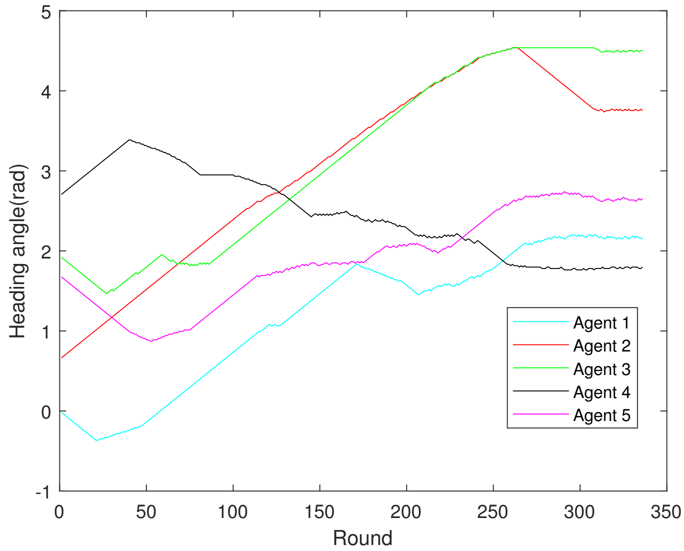

FIGURE 2.12: Changes in the heading angle of each elliptical agent

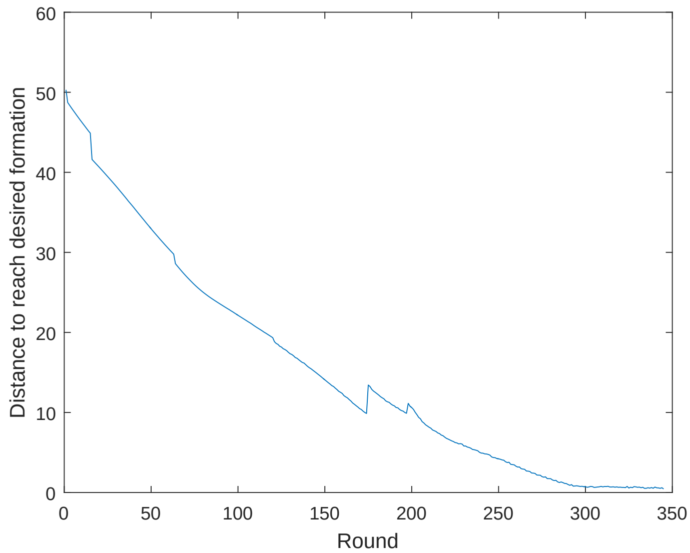

FIGURE 2.13: Distance to reach desired formation

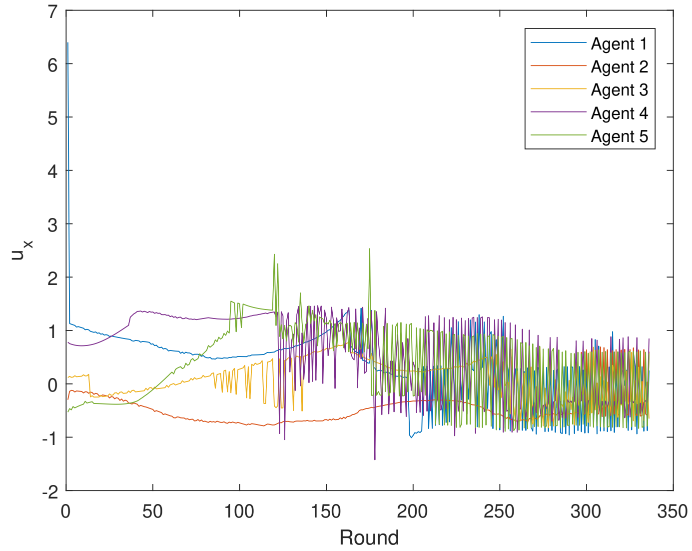

FIGURE 2.14: Control signal u_x of each elliptical agent

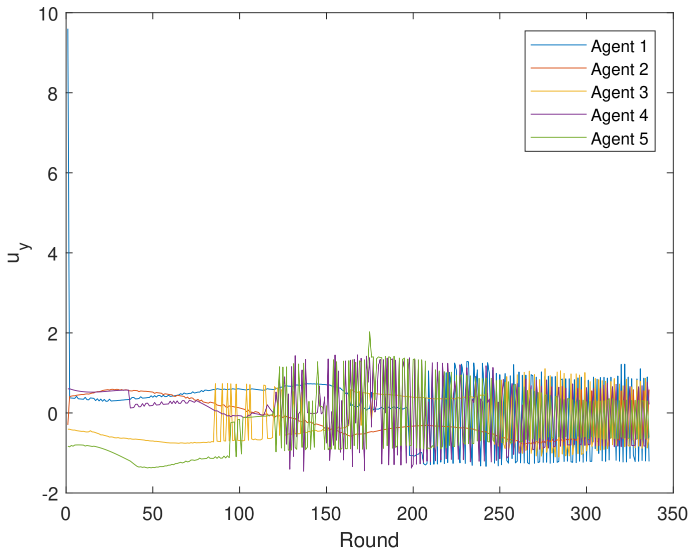

FIGURE 2.15: Control signal u_y of each elliptical agent

round. Figure 2.16 shows the trajectories of each elliptical agent in the group based on F. The final formation has the same displacements with reference formation F. It can be seen that agents move to the desired positions, which are close to them. Changes in minimum distance and heading angle of each agent in the group are shown in Figures 2.17 and 2.18, respectively. The control signals for five agents are given in Figure 2.20 and Figure 2.21. It can be seen that the heading angle change of each agent in Figure 2.12 is greater than that in Figure 2.17. This is due to the number of execution rounds. More rounds lead to a bigger change in the heading angle of each elliptical agent. Figure 2.19 displays the misalignment between temporary formation and desired formation. Different from Figure 2.13, the curve of the distance between temporary formation and desired formation drops sharply over a period time, due to the search for the optimal mapping decision. It is clearly observed from Figures 2.16-2.19 that there is no collision between any two agents. Moreover, all the agents are driven to the desired formation nicely. According to Figure 2.19 and Figure 2.20, it can be seen that the control signals are smoother when all agents are far from their desired positions, and the curves become jagged when they approach the final positions in search of a more precise formation. The communication times in each iteration of the whole system are shown in Figure 2.21. It can be seen that the communication among agents only happens when the mapping decision is changed, and only identification numbers of agents and the updated mapping decision are changed. The bandwidth can be obtained as $B = Communication\ times/Time$. In our simulation, the maximum communication times is 10. The bandwidth is 1 $Communication\ times/Time$.

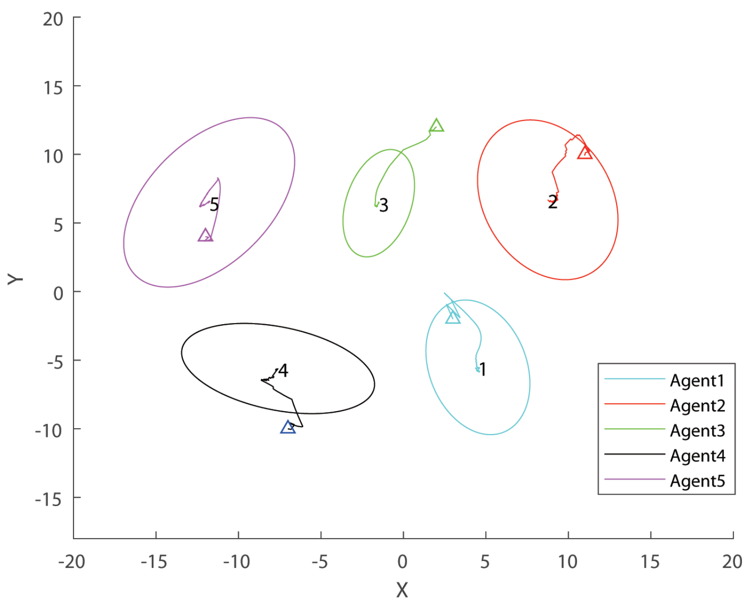

FIGURE 2.16: Trajectories of five elliptical agents

Example 3 In this example, 10 elliptical agents are employed to expand the multiple elliptical agents group. The feasibility, flexible and efficiency of the formation control strategy are illustrated in the following simulation.

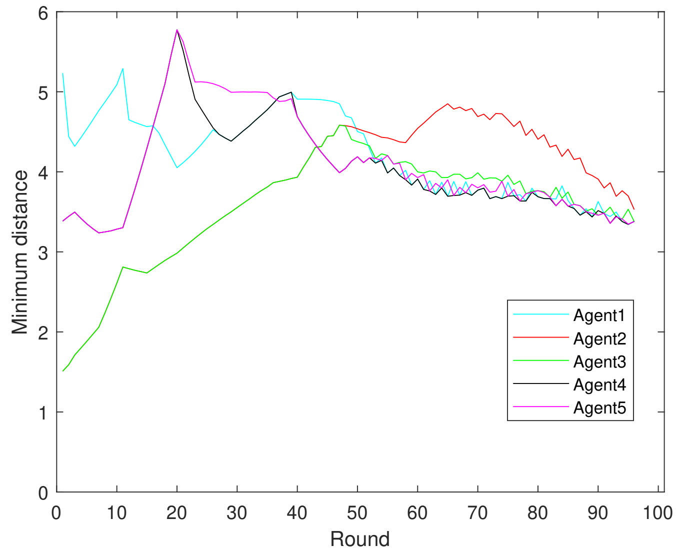

FIGURE 2.17: Changes in minimum distance of each elliptical agent

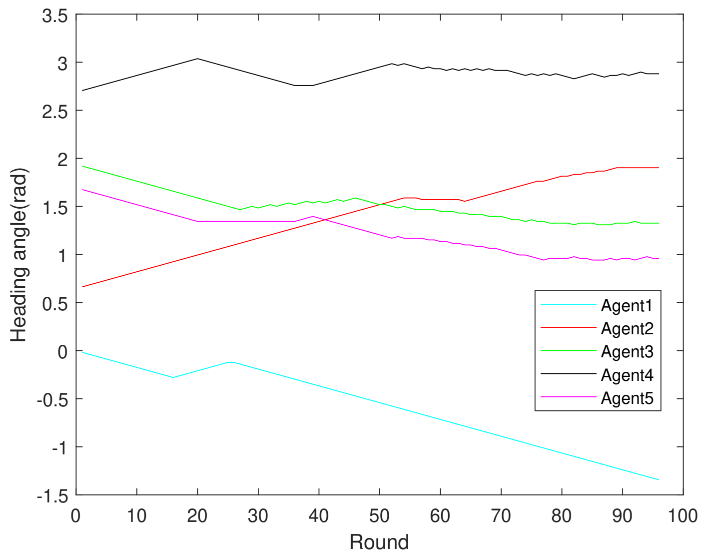

FIGURE 2.18: Changes in the heading angle of each elliptical agent

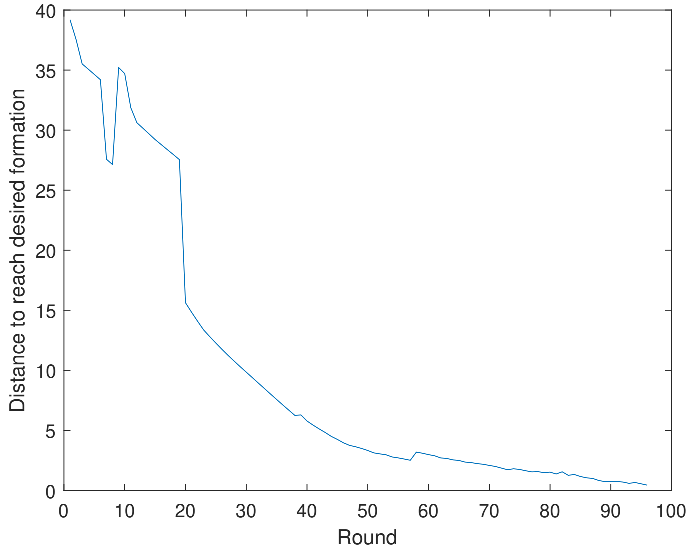

FIGURE 2.19: Distance to reach desired formation

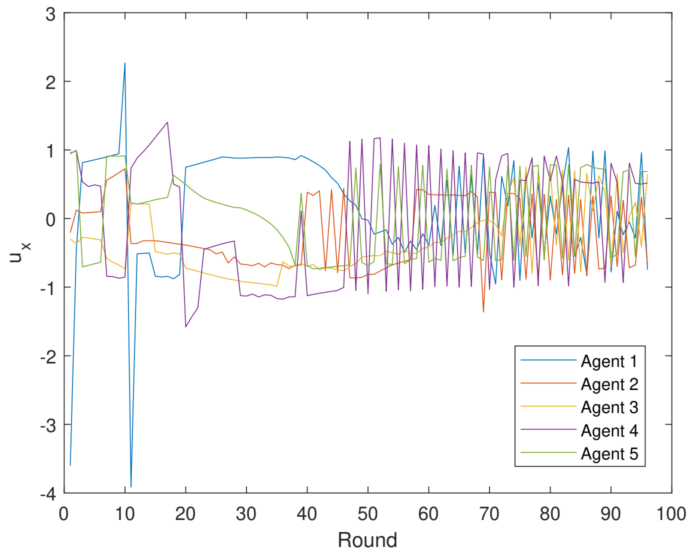

FIGURE 2.20: Control signal u_x of each elliptical agent

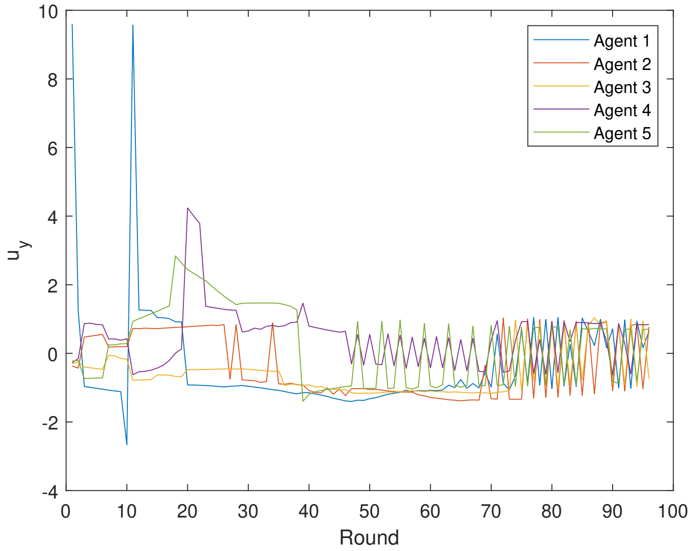

FIGURE 2.21: Control signal u_y of each elliptical agent

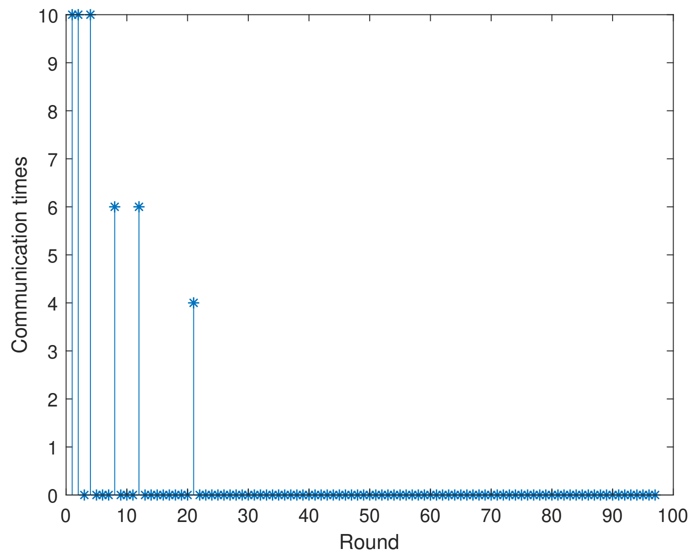

FIGURE 2.22: Communication times

First, the initial positions and heading angles of these 10 elliptical agents are given as follows.

$$\begin{bmatrix} x_{10} \\ y_{10} \\ \phi_1 \end{bmatrix} = \begin{bmatrix} 3 \\ -2 \\ 0^\circ \end{bmatrix}, \begin{bmatrix} x_{20} \\ y_{20} \\ \phi_2 \end{bmatrix} = \begin{bmatrix} 11 \\ 10 \\ 37^\circ \end{bmatrix}, \begin{bmatrix} x_{30} \\ y_{30} \\ \phi_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 15 \\ 111^\circ \end{bmatrix}, \begin{bmatrix} x_{40} \\ y_{40} \\ \phi_4 \end{bmatrix} = \begin{bmatrix} -7 \\ -10 \\ 154^\circ \end{bmatrix}, \begin{bmatrix} x_{50} \\ y_{50} \\ \phi_5 \end{bmatrix} = \begin{bmatrix} -12 \\ 4 \\ 97^\circ \end{bmatrix},$$

$$\begin{bmatrix} x_{60} \\ y_{60} \\ \phi_6 \end{bmatrix} = \begin{bmatrix} -12 \\ 15 \\ 20^\circ \end{bmatrix}, \begin{bmatrix} x_{70} \\ y_{70} \\ \phi_7 \end{bmatrix} = \begin{bmatrix} 5 \\ -10 \\ 45^\circ \end{bmatrix}, \begin{bmatrix} x_{80} \\ y_{80} \\ \phi_8 \end{bmatrix} = \begin{bmatrix} 12 \\ -2 \\ 45^\circ \end{bmatrix}, \begin{bmatrix} x_{90} \\ y_{90} \\ \phi_9 \end{bmatrix} = \begin{bmatrix} -13 \\ -11 \\ 0^\circ \end{bmatrix}, \begin{bmatrix} x_{100} \\ y_{100} \\ \phi_{10} \end{bmatrix} = \begin{bmatrix} -2 \\ 8 \\ 0^\circ \end{bmatrix}.$$

The properties of these 10 elliptical agents are given as,

$$\begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = \begin{bmatrix} 3 \\ 1 \end{bmatrix}, \begin{bmatrix} a_2 \\ b_2 \end{bmatrix} = \begin{bmatrix} 4 \\ 2 \end{bmatrix}, \begin{bmatrix} a_3 \\ b_3 \end{bmatrix} = \begin{bmatrix} 4 \\ 2 \end{bmatrix}, \begin{bmatrix} a_4 \\ b_4 \end{bmatrix} = \begin{bmatrix} 2 \\ 3 \end{bmatrix}, \begin{bmatrix} a_5 \\ b_5 \end{bmatrix} = \begin{bmatrix} 4 \\ 3 \end{bmatrix},$$

$$\begin{bmatrix} a_6 \\ b_6 \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \end{bmatrix}, \begin{bmatrix} a_7 \\ b_7 \end{bmatrix} = \begin{bmatrix} 4 \\ 1 \end{bmatrix}, \begin{bmatrix} a_8 \\ b_8 \end{bmatrix} = \begin{bmatrix} 3 \\ 1 \end{bmatrix}, \begin{bmatrix} a_9 \\ b_9 \end{bmatrix} = \begin{bmatrix} 3 \\ 2 \end{bmatrix}, \begin{bmatrix} a_{10} \\ b_{10} \end{bmatrix} = \begin{bmatrix} 2 \\ 1 \end{bmatrix}.$$

The objective map is $F = \{f_1\ f_2 \cdots f_N\}$, where

$$f_1 = \begin{bmatrix} -10 \\ 8 \end{bmatrix}, f_2 = \begin{bmatrix} 0 \\ 8 \end{bmatrix}, f_3 = \begin{bmatrix} 10 \\ 8 \end{bmatrix}, f_4 = \begin{bmatrix} -6 \\ -6 \end{bmatrix}, f_5 = \begin{bmatrix} 6 \\ -6 \end{bmatrix},$$

$$f_6 = \begin{bmatrix} 10 \\ -10 \end{bmatrix}, f_7 = \begin{bmatrix} 3 \\ 13 \end{bmatrix}, f_8 = \begin{bmatrix} -10 \\ -10 \end{bmatrix}, f_9 = \begin{bmatrix} -3 \\ 4 \end{bmatrix}, f_{10} = \begin{bmatrix} 10 \\ 4 \end{bmatrix}.$$

The initial mapping decision is give as $F_{opt} = \{3\ 1\ 2\ 5\ 4\ 7\ 6\ 8\ 9\ 10\}$. It is assumed that 10 mappings are randomly generated in each iteration. In Figure 2.23, it can be seen that agents move to the desired positions, which are close to them. Changes in minimum distance and heading angle of each agent in the group are shown in Figures 2.24 and 2.25, respectively. The control signals for the ten elliptical agents are given in Figure 2.26 and Figure 2.27. Figure 2.28 displays the misalignment between temporary formation and desired formation. It is clearly observed from Figures 2.24-2.25 that there is no collision between any two agents. Moreover, all the agents are driven to the desired formation nicely.

Note that η represents the quantity of random mapping generated per round. It can be seen from Table 2.1 that the total time and the number of rounds are different for different η. Four sets of data are listed by the average of five executions of the control algorithm. It is shown in Table 2.1 that when η increases, the number of rounds to achieve the desired formation decreases. However, time to achieve final formation will increase with the increasing η. This is because the increase of η will increase the calculating time in each execution round. Thus, it is necessary for us to choose an appropriate η to achieve the objective. In this chapter, $\eta = 5$ is optimal for 5

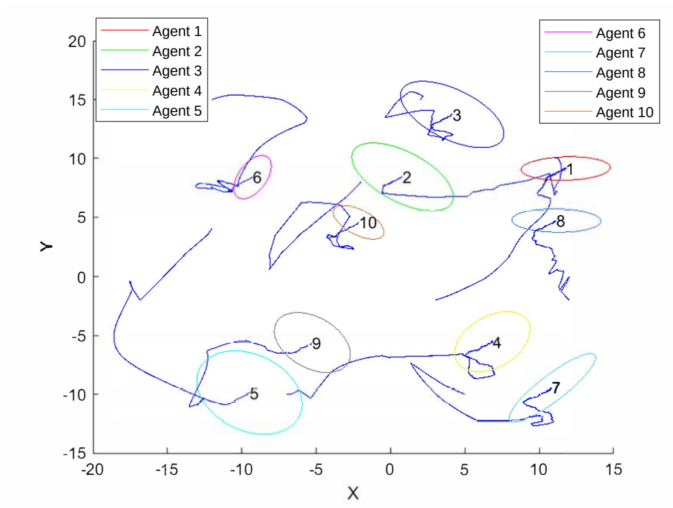

FIGURE 2.23: Trajectories of ten elliptical agents

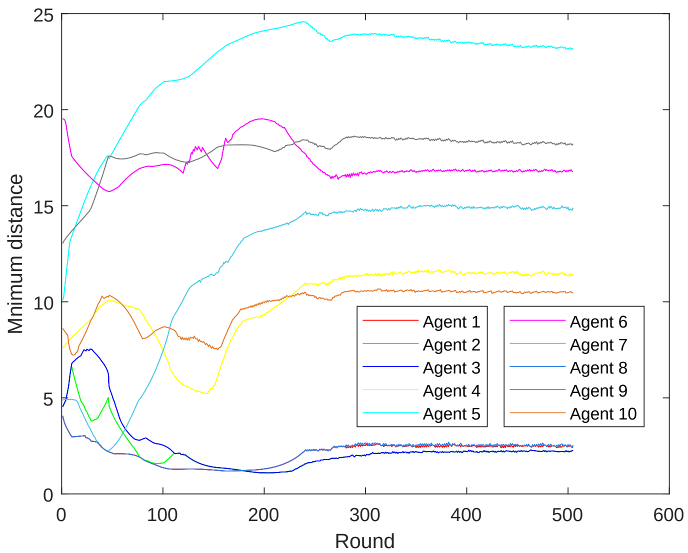

FIGURE 2.24: Changes in minimum distance of each elliptical agent

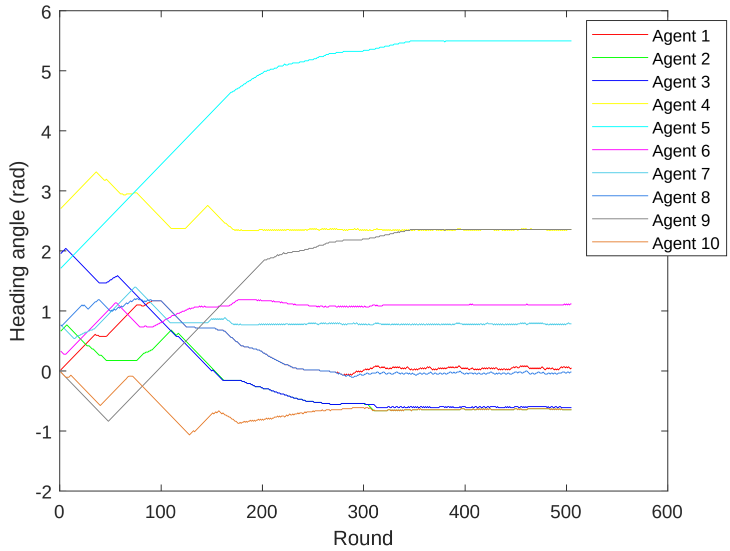

FIGURE 2.25: Changes in the heading angle of each elliptical agent

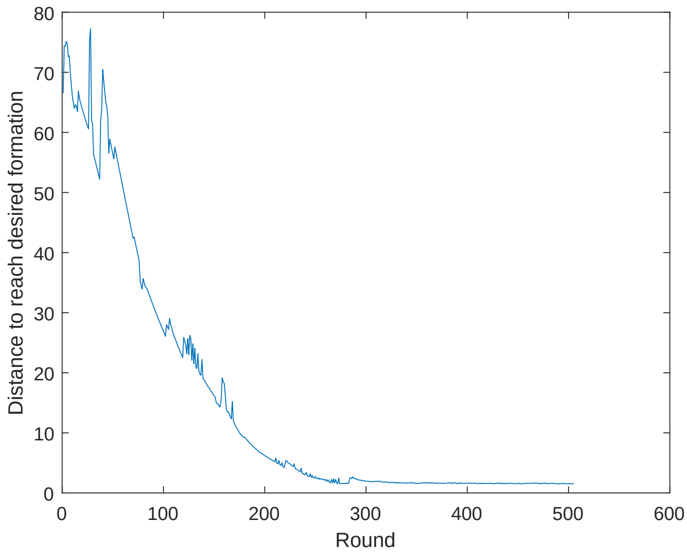

FIGURE 2.26: Distance to reach desired formation

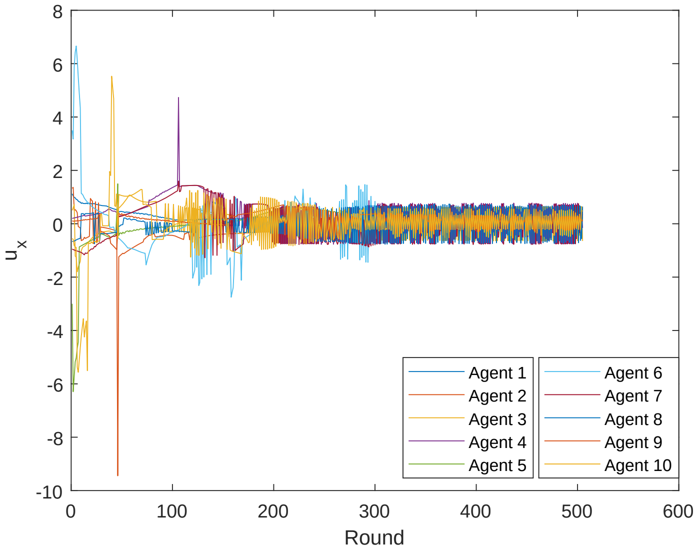

FIGURE 2.27: Control signal u_x of each elliptical agent

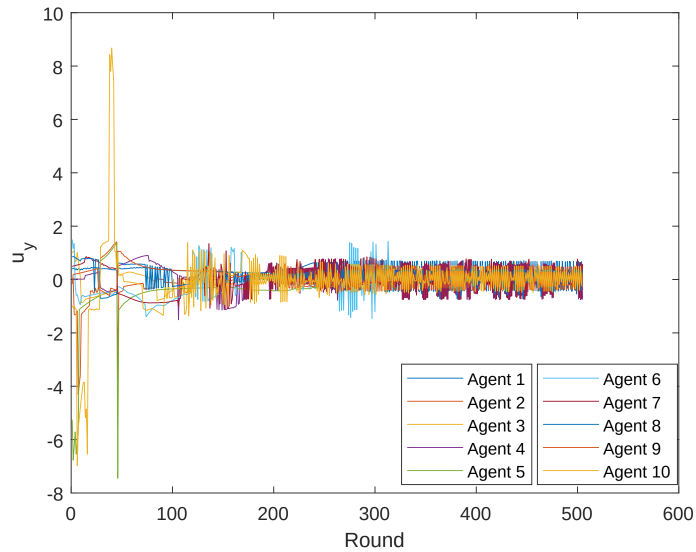

FIGURE 2.28: Control signal u_y of each elliptical agent

elliptical agents and chosen in the first two examples. The execution times of different sizes of the multi-agent system and different quantity of random mapping is given in Table 2.2 and Figure 2.29. It can be seen that the execution time substantially increases with the addition of the number of elliptical agents in the group. This is the computation burden will increase when the group is coming bigger. The value of η will also influence the computation complexity.

TABLE 2.1: Comparison of different quantity of random mapping

η	Round	Time(s)
$\eta = 0$	345	122.34
$\eta = 5$	96	102.24
$\eta = 10$	88	152.83
$\eta = 20$	81	250.02

TABLE 2.2: Comparison of different quantity of agents and η

Time(s)	$\eta = 3$	$\eta = 5$	$\eta = 10$	$\eta = 20$	$\eta = 30$
$n = 3$	32.52	43.66	-	-	-
$n = 5$	-	102.24	152.83	250.02	-
$n = 7$	-	-	427.89	567.24	846.61
$n = 10$	-	-	1255.36	3176.27	8400.27

2.5 Conclusion

This chapter proposed a control algorithm to drive a group of elliptical agents to a predefined formation based on a reference formation. The new techniques developed to achieve the objective included searching algorithms for finding minimum distance and tangents between two elliptical agents, which were used by the control algorithm. Communication among the agents was limited, and only identities of each agent and the mapping decision for them were transmitted. Random mapping algorithm was also presented to obtain optimal mapping decision for the whole group, in which a reference mapping was used to provide displacements. Collision avoidance algorithm based on collision angles between tangents, which were between agents, was used to prevent collision among agents. The self-center-based rotation algorithm for each agent was designed to further improve the collision avoidance. The simulations of fixed mapping and random mapping algorithm were given to demonstrate the feasibility and effectiveness of the new control design scheme.

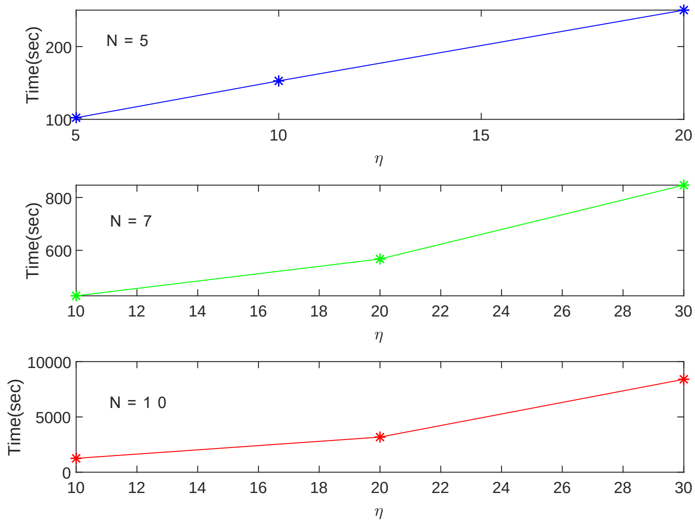

FIGURE 2.29: Comparison of different quantity of agents and η

3 Adaptive collision-free formation control for multi-agent systems

3.1 Introduction

In this chapter, an adaptive formation control scheme is constructed for a group of elliptical agents to achieve a predefined formation. The agents are assumed to have the same dynamics, and communication among the agents are limited. The desired formation is realized based on the reference formation and the mapping decision. In the control design, searching algorithms for both cases of minimum distance and tangents are established for each agent and its neighbors. In order to avoid collision, an optimal path planning algorithm based on collision angles, and a self-center-based rotation algorithm are also proposed. Moreover, randomized method is used to provide the optimal mapping decision for the underlying system.

Different from the work in [94], [95], this chapter focuses on driving a group of agents to achieve a desired formation derived from the objective map. The objective map is set well in advance and serves as a reference. Agents only organise their formation based on the displacements in the objective map, but not the specific points in the predefined map. The obstacle avoidance algorithm established in this chapter is based on the optimal path planning by removing the obstacle angles. These angles can be obtained by clamping tangents of objective agents and their obstacle agents. To improve the efficiency of the collision-free formation control algorithm for multiple elliptical agents, an adaptive parameter is introduced, and dynamic mapping algorithm is employed to enhance the efficiency of the whole group.

In this chapter, the main work is as follows. First, a new control scheme is proposed to drive a group of elliptical agents to a predefined formation. All agents are assumed to have the same form of control law and reference formation. Only restricted communication among agents is allowed, and they can send and receive identification numbers to and from other agents in the system. The controller of each agent is established based on the midpoint derived from their neighbourhood. Second, the predefined formation is based on the displacements, which are obtained though a reference mapping. Agents can find their optimal mapping decisions based on the random mapping algorithm. During each sampling interval, several possible mappings are generated and the sums of distances with corresponding agents under each possible mapping decision are calculated to be compared with the others. The shortest one will be chosen to be the optimal formation in the corresponding interval. Third, the collision among elliptical agents can be avoided by choosing optimal path and removing obstacle angles. A self-center-based rotation algorithm is also proposed to guarantee collision avoidance when two agents approach to each other.

3.2 System model

Consider N agents that form the multi-agent system, and agents are described as ellipses. The elliptical formula of agent E_i is given as

$$E_i : [x_i, y_i, 1]A_i[x_i, y_i, 1]^T = 0 \tag{3.1}$$

where A_i is the parameter matrix based on heading angle ϕ_i

$$A_i = \begin{bmatrix} \frac{\cos^2\phi_i}{a_i^2} + \frac{\sin^2\phi_i}{b_i^2} & \frac{\sin 2\phi_i}{a_i^2} & \frac{2A_{i1}\cos\phi_i}{a_i^2} \\ -\frac{\sin 2\phi_i}{b_i^2} & \frac{\sin^2\phi_i}{a_i^2} + \frac{\cos^2\phi_i}{b_i^2} & \frac{2A_{i1}\sin\phi_i}{a_i^2} \\ -\frac{2A_{i2}\sin\phi_i}{b_i^2} & \frac{2A_{i2}\cos\phi_i}{b_i^2} & \frac{A_{i1}^2}{a_i^2} + \frac{A_{i2}^2}{b_i^2} - 1 \end{bmatrix},$$

and

$$A_{i1} = x_{i0}\cos\phi_i + x_{i0}\sin\phi_i,$$
$$A_{i2} = -y_{i0}\sin\phi_i + y_{i0}\cos\phi_i.$$

The point (x_i, y_i) on E_i can be obtained from (3.1), while a_i and b_i are the long and short axes of E_i. The coordinate of center point p_i^c of E_i is represented by (x_{i0}, y_{i0}). The set of the agents is given by

$$E = \{E_1\ E_2\ E_3\ \dots\ E_N\}.$$

The dynamics of elliptical agent E_i is described by a single integrator model,

$$\dot{x}_i^c = u_i. \tag{3.2}$$

For all $E_i \in E$, the vector $u_i = [u_{xi}, u_{yi}]^T$ is the position control input. For any point p_i on agent E_i, and for any point p_j on agent E_j, at the initial time t_0, the distance between these two points satisfies

$$|x_i - x_j| > \varepsilon, \tag{3.3}$$

where $\varepsilon > 0$, and ε should be big enough to make sure collision-free among elliptical agents. If all elliptical agents satisfy (3.3), superpositions among agent will not occur at the initial time t_0.

For agent E_i, the sensing range is an oval, and the center of sensing ellipse is at p_i^c, while the long axis is given as $a_i^{sen} = a_i + R_{sen}$ and the short axis is given as $b_i^{sen} = b_i + R_{sen}$. The sensing range is R_{sen}. If a point $p = (x, y)$ satisfies

$$\begin{aligned} &[x, y, 1]A_i[x, y, 1]^T > 0, \\ &[x, y, 1]B_i[x, y, 1]^T < 0, \end{aligned} \tag{3.4}$$

then p can be sensed by E_i. In (4),

$$B_i = \begin{bmatrix} \frac{\cos^2\phi_i}{(a_i^{sen})^2} + \frac{\sin^2\phi_i}{(b_i^{sen})^2} & \frac{\sin 2\phi_i}{(a_i^{sen})^2} & \frac{2B_{i1}\cos\phi_i}{(a_i^{sen})^2} \\ -\frac{\sin 2\phi_i}{(b_i^{sen})^2} & \frac{\sin^2\phi_i}{(a_i^{sen})^2} + \frac{\cos^2\phi_i}{(b_i^{sen})^2} & \frac{2B_{i1}\sin\phi_i}{(a_i^{sen})^2} \\ -\frac{2B_{i2}\sin\phi_i}{(b_i^{sen})^2} & \frac{2B_{i2}\cos\phi_i}{(b_i^{sen})^2} & B_{i3} \end{bmatrix},$$

$$\begin{aligned} B_{i1} &= x_{i0}\cos\phi_i + y_{i0}\sin\phi_i, \\ B_{i2} &= -x_{i0}\sin\phi_i + y_{i0}\cos\phi_i, \\ B_{i3} &= \frac{B_{i1}^2}{(a_i^{sen})^2} + \frac{B_{i2}^2}{(b_i^{sen})^2} - 1. \end{aligned}$$

The surface of avoidance range for E_i is also an oval, which is centered at p_i^c, while the long axis is given as $a_i^{avo} = a_i + R_{avo}$ and the short axis is given as $b_i^{avo} = b_i + R_{avo}$.

Some assumptions are imposed on the sensing ability and initial position conditions.

Assumption 5.

For system (3.2), the following conditions are satisfied.

(i) Agents can move in any 2D directions.

(ii) Agents can figure out the center of other agents, at least a portion of which is within their sensing ranges.

Remark 3. *Condition (i) in Assumption 5 shows that there is no limitation for each agent to move. All agents can choose the optimal moving direction without rotation. Based on Condition (ii), if any point p_j on agent E_j satisfies (3.4), E_j can be seen as the neighboring agent of E_i. The neighborhood $H_{E_i}(R_{sen})$ of E_i can be written as*

$$H_{E_i}(R_{sen}) = \{E_1^{H_{E_i}(R_{sen})}\ E_2^{H_{E_i}(R_{sen})}\ \cdots\}.$$

Also, the avoidance group $A_{E_i}(R_{sen})$ of E_i can be written as

$$A_{E_i}(R_{avo}) = \{E_1^{A_{E_i}(R_{avo})}\ E_2^{A_{E_i}(R_{avo})}\ \cdots\}.$$

3.3 Adaptive controller design

In this section, an adaptive controller is proposed to achieve the predefined formation. Reference formation F is given as

$$F = \{f_1\ f_2\ \cdots\ f_N\},$$

where $f_i \in F$ is the coordinate of ith position in the reference formation. The optimal mapping in the kth iteration is given as $r_{op}(k)$. If agent E_i moves without possible collision, control input $u_i(k)$ in the kth iteration for E_i can be obtained by

$$u_i(k) = \sigma(g_i(k) - p_i^c(k)), \tag{3.5}$$

where $p_i^c(k)$ is the center of agent E_i in the kth iteration, while $g_i(k)$ is the desired position in the kth iteration based on $R_{op}(k)$. The coefficient σ represents the control step. The desired position $g_i(k)$ can be written as

$$g_i(k) = f_i(R_{op}(k)) + (O_i - O_{f_i(R_{op}(k))}). \tag{3.6}$$

In (3.6), $f_i(R_{op}(k))$ is the position corresponding to predefined formation F based on $R_{op}(k)$. The center point $O_{f_i(R_{op}(k)})$ is the center of desired position of agent E_i and the desired positions of its neighboring agents which correspond to F with the mapping rule $R_{op}(k)$. We can have $O_{f_i(R_{op}(k)})$ as

$$O_{f_i(R_{op}(k))} = \frac{f_i(R_{op}(k)) + \sum_{j=1}^{m} f_j^i(R_{op}(k))}{m+1},$$

where $f_j^i(R_{op}(k))$ is the desired position of agent E_j based on $R_{op}(k)$, while $E_j \in H_{E_i}(R_{sen})$. The number of elements in $H_{E_i}(R_{sen})$ is m. The center of current positions of E_i and its neighboring agents is O_i, which can be obtained by

$$O_i = \frac{x_i^c(k) + \sum_{j=1}^{m} x_j^c(k)}{m+1},$$

where $p_i^c(k)$ is the current center position of agent E_i in the kth iteration, while $p_j^c(k)$ is the current center position of agent $E_j \in H_{E_i}(R_{sen})$. The controller for agent E_i with two neighbors is given in Figure 3.1, and the sensing range R_{sen} is also shown.

If agent E_i moves though the possible collision area, the controller should be updated. The possible collision area is based on the avoidance group $A_{E_i}(R_{avo})$. The angle $\psi_{ij}(k)$ between E_i and $E_j \in A_{E_i}(R_{avo})$ and the angle $\varphi_{ij}(k)$ between $u_i(k)$ and midline $m_{ij}(k)$ of $\psi_{ij}(k)$ should be calculated to judge whether agent E_i moves though possible collision area or not. They can be obtained from

$$\psi_{ij}(k) = \arccos\left(\frac{(p_j^{l_1^{ij}}(k) - p_i^{l_1^{ij}}(k)) \cdot (p_j^{l_2^{ij}}(k) - p_i^{l_2^{ij}}(k))}{|p_j^{l_1^{ij}}(k) - p_i^{l_1^{ij}}(k)||p_j^{l_2^{ij}}(k) - p_i^{l_2^{ij}}(k)|}\right), \tag{3.7}$$

$$\varphi_{ij}(k) = \arccos\left(\frac{u_i(k) \cdot u_j(k)}{|u_i(k)||u_j(k)|}\right). \tag{3.8}$$

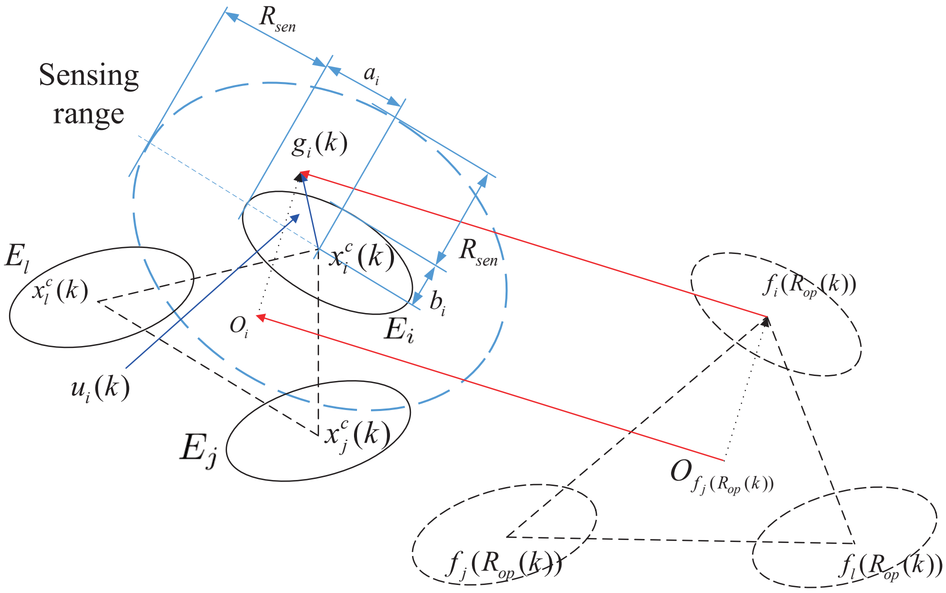

FIGURE 3.1: Controller and sensing range for agent E_i with two neighbors in the kth iteration

In (7), l_1^{ij} and l_2^{ij} are the two cross tangents of E_i and $E_j \in A_{E_i}(R_{avo})$ at the kth iteration. The corresponding points on E_i and E_j based on l_1^{ij} and l_2^{ij} are $p_i^{l_1^{ij}}(k)$, $p_j^{l_1^{ij}}(k)$, $p_i^{l_2^{ij}}(k)$ and $p_j^{l_2^{ij}}(k)$. The control input of E_j is $u_j(k)$. If $\varphi_{ij}(k) > \psi_{ij}(k)$ exists, then agent E_i will move though possible collision area. Hence control input $u_i(k)$ should be updated as

$$u_i^d(k) = \sigma |u_i^d(k)| \frac{(p_i^{l_o^{il}}(k), p_l^{l_o^{il}}(k))}{|(p_i^{l_o^{il}}(k), p_l^{l_o^{il}}(k))|}, o = 1, 2 \tag{3.9}$$

$$|u_i^d(k)| = \xi_i(k)(d_{il}^{min}(k) - d_{stop}), \tag{3.10}$$

$$g_i(k) = p_i^c(k) + \frac{u_i^d(k)}{\sigma}. \tag{3.11}$$

where $E_l \in A_{E_i}(R_{avo})$ satisfying $\varphi_{il}(k)$ is the minimum among the avoidance group, and d_{il}^{min} is the minimum distance value between E_i and E_l. The desired position of E_i is updated in (3.11). The anti-collision parameter is given as d_{stop}, which is a positive constant. The coefficient $\xi_i(k)$ is related to the distance $d_i(k)$ between $p_i^c(k)$ and $g_i(k-1)$ and the sum of the minimum distances between E_i and agents belonging to $A_{E_i}(R_{avo})$. Distance $d_i(k)$ is given as

$$d_i(k) = |g_i(k-1) - p_i^c(k)|.$$

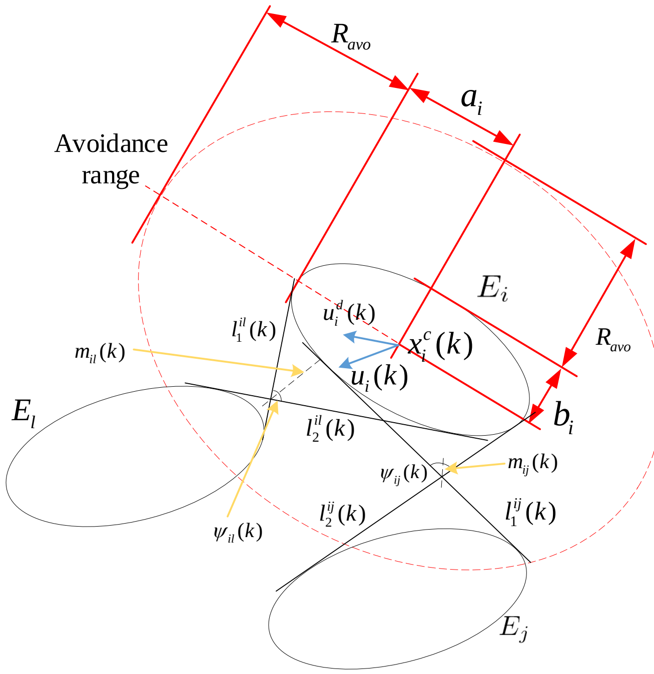

FIGURE 3.2: Collision-free controller and avoidance range for agent E_i in the kth iteration

Coefficient $\xi_i(k)$ is written as

$$\xi_i(k) = \lambda \frac{d_i(k)}{\sum_{j=1}^{p} d_{ij}^{min}(k)}. \tag{3.12}$$

The crossed tangents between E_i and E_l is given as $l_o^{il}, o = 1, 2$, while p is the number of the elements in $A_{E_i}(R_{avo})$. The updated control strategy is illustrated in Figure 3.2.

Theorem 1. *Consider system (3.2) under Assumption 5, if*

$$0 < \lambda < \frac{(n-1)R_{avo}(1 + \frac{d_{stop}}{\varepsilon - d_{stop}})}{d_i(k) + d_l(k)},$$

then the following inequality holds:$|x_i(k) - x_l(k)| > \varepsilon$*, where* $p_i(k)$ *and* $p_l(k)$ *represent any point on* E_i *and* E_l *at the kth iteration, respectively, and* $\varepsilon > 0$*. Agent* E_l *satisfies (3.9).*

Proof. In order to have $|p_i(k) - p_l(k)| > \varepsilon$, it is sufficient to have $d_{il}^{min}(k) > \varepsilon$. To construct the collision-free system, we should have

$$0 < |u_i(k)| + |u_l(k)| \leq d_{il}^{min}(k). \tag{3.13}$$

Based on (3.10), (3.13) can be written as,

$$0 < (\xi_i(k) + \xi_l(k))(d_{il}^{min}(k) - d_{stop}) \leq d_{il}^{min}(k),$$

$$0 < \xi_i(k) + \xi_l(k) \leq \frac{d_{il}^{min}(k)}{(d_{il}^{min}(k) - d_{stop})}. \tag{3.14}$$

Substituting (3.12) into (3.14) yields,

$$0 < \lambda \frac{d_i(k)}{\sum_{j=1}^{b} d_{ij}^{min}(k)} + \lambda \frac{d_l(k)}{\sum_{j=1}^{q} d_{lj}^{min}(k)} \leq \frac{d_{il}^{min}(k)}{(d_{il}^{min}(k) - d_{stop})}, \tag{3.15}$$

where b and q are the number of agents in $A_{E_i}(R_{avo})$ and $A_{E_l}(R_{avo})$, respectively. The minimum distances for E_i and E_l with their avoidance group are given as $d_{ij}^{min}(k)$, $d_{lj}^{min}(k)$, respectively. Let $d = |d_{il}^{min}(k)| - d_{stop}$, we have $|d_{il}^{min}(k)| = d + d_{stop} > \varepsilon$, (15) can be written as

$$0 < \lambda \frac{d_i(k)}{\sum_{j=1}^{b} d_{ij}^{min}(k)} + \lambda \frac{d_l(k)}{\sum_{j=1}^{q} d_{lj}^{min}(k)} < 1 + \frac{d_{stop}}{\varepsilon - d_{stop}}. \tag{3.16}$$

It can be seen that $1 \leq b \leq n-1$ and $1 \leq q \leq n-1$, while $d_{ij}^{min} \leq R_{avo}$ and $d_{lj}^{min} \leq R_{avo}$. Then, we have $\sum_{j=1}^{b} d_{ij}^{min}(k) \leq (n-1)R_{avo}$ and $\sum_{j=1}^{b} d_{ij}^{min}(k) \leq (n-1)R_{avo}$.

$$\frac{d_i(k)}{\sum_{j=1}^{b} d_{ij}^{min}(k)} \geq \frac{d_i(k)}{(n-1)R_{avo}}, \tag{3.17}$$

$$\frac{d_l(k)}{\sum_{j=1}^{b} d_{lj}^{min}(k)} \geq \frac{d_l(k)}{(n-1)R_{avo}}. \tag{3.18}$$

Hence, substituting (3.17) and (3.18) into (3.16) gives

$$0 < \lambda \frac{d_i(k)}{(n-1)R_{avo}} + \lambda \frac{d_l(k)}{(n-1)R_{avo}} < 1 + \frac{d_{stop}}{\varepsilon - d_{stop}} \tag{3.19}$$

The range of λ can be calculated as

$$0 < \lambda < \frac{(n-1)R_{avo}(1 + \frac{d_{stop}}{\varepsilon - d_{stop}})}{d_i(k) + d_l(k)}. \tag{3.20}$$

If λ satisfies (3.20), the condition $|p_i(k) - p_l(k)| > \varepsilon$ holds, which means collision will not occur on agent E_i. This completes the proof. □

3.4 Adaptive random mapping algorithm

Mapping is the rule that connects predefined formation F and desired position set $G = \{g_1\ g_2 \cdots g_N\}$ of all agents in the multi-agent systems.To obtain the optimal mapping of the multiple elliptical agents, a novel adaptive random mapping algorithm is developed. The steps of the adaptive random mapping algorithm are given as follows:
Initialization: In the first κ iterations, a random mapping algorithm is employed to find the optimal mapping in each iteration. In each iteration, η mappings are generated, and the set of η mappings and each mapping are given as

$$
\begin{aligned}
R(k) =& \{r_1(k)\ r_2(k) \cdots r_\eta(k)\}, 0 < k \leq \kappa, \\
r_s(k) =& \{r_s(k)(E_1)\ r_s(k)(E_2) \cdots r_s(k)(E_N)\}, 0 < s \leq \eta.
\end{aligned}
$$

Here, $r_s(k)$ is the sth mapping, and the elements in $r_s(k)$ are random generated from 1 to n, while any two elements in $R(k)$ are not equal. The total distance $L_s(k)$ of the sth mapping can be obtained based on the current positions and the desired positions in kth iteration of all agents, and can be obtained as

$$
L_s(k) = \sum_{i=1}^{N} |g_i(k) - p_i^c(k)|. \tag{3.21}
$$

The optimal mapping $r_{op}(k)$ in the kth iteration satisfies

$$
L_{op}(k) = \min\{L_1(k)\ L_2(k) \cdots L_\eta(k)\ L_{op}(k-1)\},
$$

while $r_{op}(k) \in R(k)$. Here, $L_{op}(k-1)$ represents the total distance of the optimal mapping $r_{op}(k-1)$ in the $(k-1)$th iteration.
Then: When $k > \kappa$, the screening group $R_f(k)$ is constructed based on the optimal mappings from $(k-\kappa)$th to kth iterations, which can be written as

$$
R_f(k) = \{r_{op}(k-\kappa)\ r_{op}(k-\kappa+1) \cdots r_{op}(k-1)\}, k > \kappa.
$$

The repeat mappings in $R_f(k)$ should be extracted $R_r(k) = \{r_1^r\ r_2^r \cdots\}$, while $r_i^r \in R_r(k)$ also satisfies $r_i^r \in R_f(k)$. The invariant elements which are in the same positions in different mapping in $R_r(k)$ should be found. Then these elements are specified as fixed value in $r_{op}(k)$. The values of the other elements should be regenerated. If t elements are different in each mapping, t mappings will be regenerated based on the invariant elements. Optimal mapping $r_{op}(k)$ will be obtained by the minimum value of corresponding total distances of t mappings based on (3.21). If all mappings in $R_f(k)$ are the same, then $r_{op}(k) = r_{op}(k-1)$.

The proposed adaptive random mapping algorithm can release the computation burden. This mapping algorithm will be terminated when desired formation is established. The flowchart of the adaptive mapping algorithm is given in Figure 3.3.

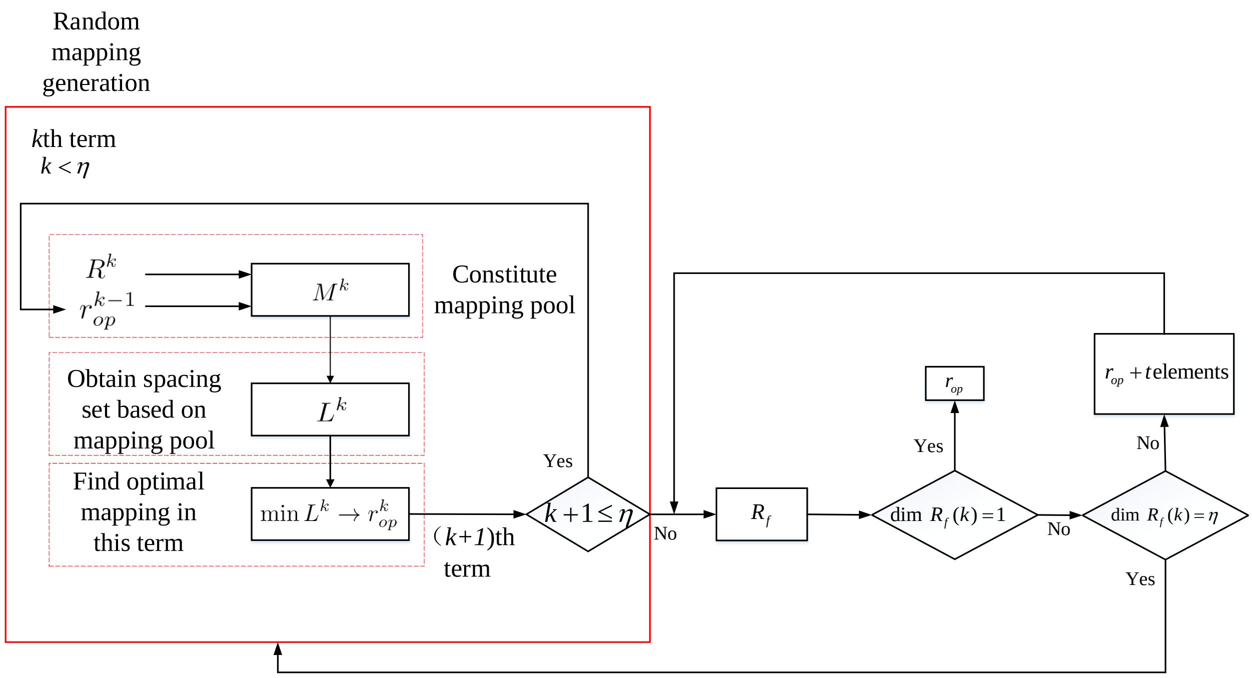

FIGURE 3.3: Adaptive random mapping algorithm

3.5 Simulation results

In this section, simulation results are presented to illustrate the effectiveness of the proposed algorithm. In the simulation, five elliptical agents form the multi-agent system. All agents have different long axis and short axis, which are given as

$$\begin{bmatrix} a_1 \\ b_1 \end{bmatrix} = \begin{bmatrix} 5 \\ 3 \end{bmatrix}, \begin{bmatrix} a_2 \\ b_2 \end{bmatrix} = \begin{bmatrix} 3 \\ 1 \end{bmatrix}, \begin{bmatrix} a_3 \\ b_3 \end{bmatrix} = \begin{bmatrix} 4 \\ 2 \end{bmatrix},$$

$$\begin{bmatrix} a_4 \\ b_4 \end{bmatrix} = \begin{bmatrix} 5 \\ 2 \end{bmatrix}, \begin{bmatrix} a_5 \\ b_5 \end{bmatrix} = \begin{bmatrix} 6 \\ 4 \end{bmatrix}.$$

The initial position and heading angle of each agent are given as follows

$$\begin{bmatrix} x_{10} \\ y_{10} \\ \phi_1 \end{bmatrix} = \begin{bmatrix} 3 \\ -2 \\ 0^\circ \end{bmatrix}, \begin{bmatrix} x_{20} \\ y_{20} \\ \phi_2 \end{bmatrix} = \begin{bmatrix} 10 \\ 10 \\ 37^\circ \end{bmatrix}, \begin{bmatrix} x_{30} \\ y_{30} \\ \phi_3 \end{bmatrix} = \begin{bmatrix} 2 \\ 12 \\ 100^\circ \end{bmatrix},$$

$$\begin{bmatrix} x_{40} \\ y_{40} \\ \phi_4 \end{bmatrix} = \begin{bmatrix} -7 \\ -10 \\ 154^\circ \end{bmatrix}, \begin{bmatrix} x_{50} \\ y_{50} \\ \phi_5 \end{bmatrix} = \begin{bmatrix} -12 \\ 4 \\ 97^\circ \end{bmatrix}.$$

The sensing range is $R_{sen} = 4$. The avoidance range is $R_{avo} = 2$. The control step is $\sigma = 0.1$. The initial iterations κ is given as $\kappa = 6$. The anti-collision parameter d_{stop} in (3.10) is 0.1. The parameter ε in (3.16) is given as $\varepsilon = 0.5$. The predefined formation is set to $F = \{[-10, 8]; [0, 4]; [8, 8]; [-6, -4]; [6, -4]\}$. Parameter λ in (3.12) is given as $\lambda = 0.5$. The formation process is given in Figure 3.4. For the legend $*$ used in Figure 3.4, it denotes a number that is the identity of an elliptical agent. The legend Δ used in Figure 3.4, denotes the initial position of each elliptical agent. The minimum distance between each

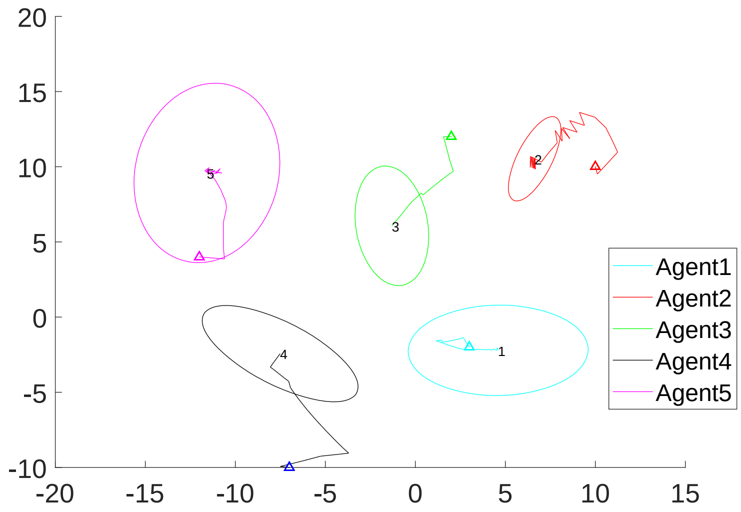

FIGURE 3.4: Trajectories of the five elliptical agents

elliptical agent and its nearest neighbor is given in Figure 3.5, which illustrates that there is no collision among the agents during moving. The control

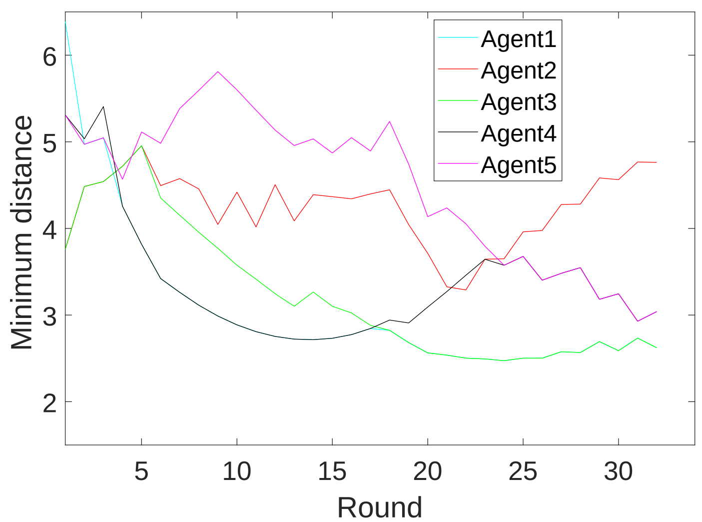

FIGURE 3.5: Minimum distance between each elliptical agent and its nearest neighbor

input for each elliptical agent is shown in Figure 3.6. All elliptical agents can move in any direction in 2D space. The parameter ξ in (3.10) is adaptive based on the distance between current position and desired position of each agent and the sum distance of minimum distances between agents and the agents in their avoidance group. The changes of ξ during moving is given in Figure 3.7. In the figures, $\xi = 0$ implies that there is no possibility of collision of corresponding agent.

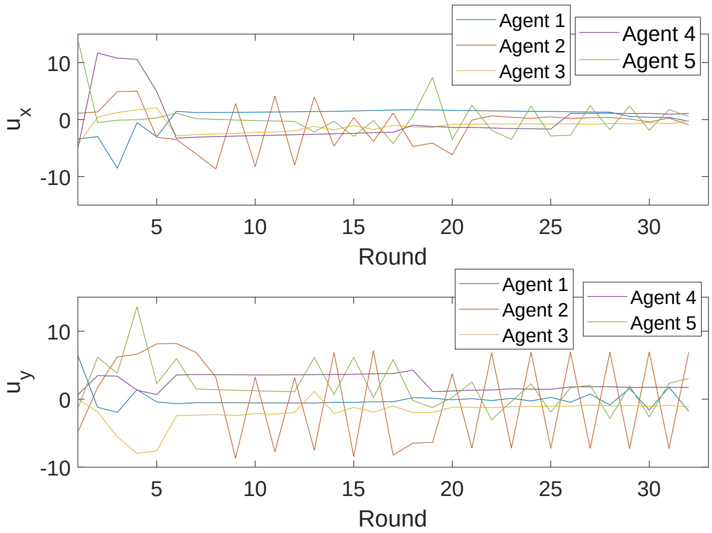

FIGURE 3.6: Control inputs for five elliptical agents

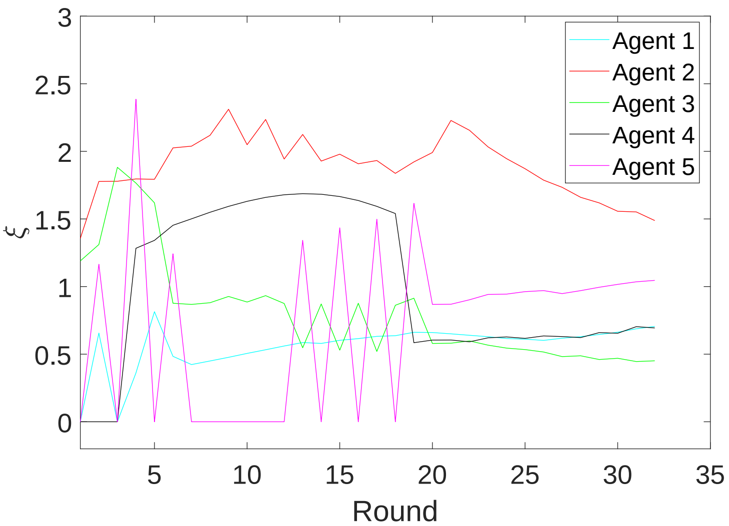

FIGURE 3.7: Changes of collision-free control parameter ξ

The comparison of the adaptive random mapping algorithm and the random mapping algorithm in [162] based on communication times is shown in Figure 3.8. It can clearly be seen that the communication times decrease, which implies the effectiveness of proposed adaptive random mapping algorithm.

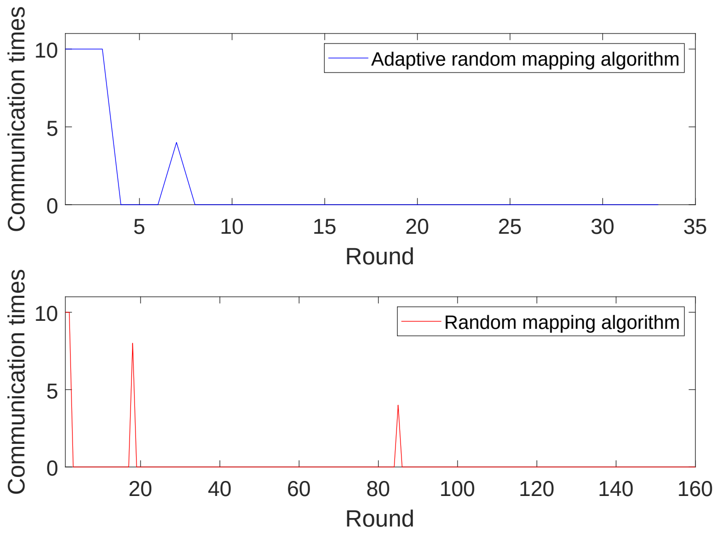

FIGURE 3.8: Comparison of the adaptive random mapping algorithm and the random mapping algorithm in [162]

3.6 Conclusion

This chapter established a formation control strategy to enable a group of elliptical agents to achieve predefined formation. The control input of each agent is based on the displacements between the agent and its neighboring agents. To build the collision-free control strategy, the avoidance group of each agent based on the avoidance range and minimum distances among elliptical agents are employed. An adaptive random mapping algorithm is proposed for obtaining the optimal mapping decision. Simulation results show the feasibility and effectiveness of the novel control strategy.

4 Event-triggered probability-driven adaptive formation control for multiple elliptical agents

4.1 Introduction

In this chapter, an event-triggered control scheme is proposed for a group of elliptical agents to achieve a predefined formation. The agents are assumed to have the same dynamics. The control law for each agent is only updated at its event sequence based on its own minimum collision time and deviation time. The probability-driven controller is designed to prevent the stuck problem among agents. Mapping-adaptive strategy and angle-adaptive scheme based on the minimum collision distance are also developed. Two examples with analysis are presented to demonstrate the effectiveness and potential of the new event-triggered adaptive control algorithm.

Some formation controllers are investigated for multiple elliptical agents. In [162], [163], a controller for multiple elliptical agents to reach predefined formation is proposed with limited sensing range. The formation can be constructed in any space with a reference formation, and a collision avoidance algorithm with collision angle is employed for each agent. Communication topology of multi-agent systems is studied in [164]–[166], in which the information of each agent broadcasts to its neighbors to accomplish the control process. Unlike the aforementioned research, this chapter investigates the problem of formation control for a group of agents with elliptical shape, and design a probability-driven controller based on event-triggered method and adaptive strategies. Under such a scheme, each agent will move along its planned path and change only happens when pre-defined event conditions are triggered. Each individual agent can only obtain the information from other agents through communication among agents when the control law of itself is updated. In this process, an adaptive method is employed to control the heading angles of agents to minimize the probability of collision among the agents, and to adjust the mapping of the system to reduce the moving distance of each agent. Moreover, without loss of generality, all the agents are assumed to have the same dynamics and to be equally important.

In this chapter, the main work is as follows. First, each agent has its own event sequence based on the minimum collision time and deviation time calculated by itself. Agents only need to receive the state and velocity information in accordance with their own event sequence. Then, probability-driven controller is established to prevent the stuck problem among agents, which may happen when two or more elliptical agents are too close to each other. Also, heading angle rotation algorithm is employed to extend the minimum collision distance for each agent to further prevent collisions among agents. Adaptive mapping scheme is also given to shrink the distance between current positions of elliptical agents and the predefined formation and action time of the whole system.

Notation. Throughout this chapter, $\mathbf{R}^m$ is an mth dimensional space of real numbers. The symbol $|\cdot|$ represents the length of a vector. For any two points a and b, $\overrightarrow{(a,b)}$ represents the vector between a and b. If dimensions of matrices are not explicitly stated, they are assumed to be compatible for algebraic operations.

4.2 Problem statements

In this chapter, the agents concerned in formation control are assumed to have the shape of ellipses. The relevant elliptical formulas will be given first followed by the problem statement.

4.2.1 Ellipse formula

Consider the ith elliptical agent $E_i \in \mathbf{R}^2$ of which heading angle is ϕ_i, centered at $(x_{i0}, y_{i0})^T$, the elliptical expression can be given as,

$$E_i : [x_i, y_i, 1]A_i[x_i, y_i, 1]^T = 0 \tag{4.1}$$

where

$$A_i = \begin{bmatrix} \frac{\cos^2\phi_i}{a_i^2} + \frac{\sin^2\phi_i}{b_i^2} & \frac{\sin 2\phi_i}{a_i^2} & \frac{2A_{i1}\cos\phi_i}{a_i^2} \\ -\frac{\sin 2\phi_i}{b_i^2} & \frac{\sin^2\phi_i}{a_i^2} + \frac{\cos^2\phi_i}{b_i^2} & \frac{2A_{i1}\sin\phi_i}{a_i^2} \\ -\frac{2A_{i2}\sin\phi_i}{b_i^2} & \frac{2A_{i2}\cos\phi_i}{b_i^2} & \frac{A_{i1}^2}{a_i^2} + \frac{A_{i2}^2}{b_i^2} - 1 \end{bmatrix},$$

where A_i is the coefficient matrix based on heading angle ϕ_i with

$$\begin{aligned} A_{i1} &= x_{i0}\cos\phi_i + y_{i0}\sin\phi_i, \\ A_{i2} &= -x_{i0}\sin\phi_i + y_{i0}\cos\phi_i. \end{aligned}$$

In (4.1), parameter a_i represents the long axis of elliptical agent E_i, and parameter b_i is the short axis. The set of points on E_i can be described as follow,

$$\begin{aligned} P_i = \{p_i = (x_i, y_i)| \\ & x_i = a_i\cos\theta\cos\phi_i - b_i\sin\theta\sin\phi_i + x_{i0}, \\ & y_i = a_i\cos\theta\sin\phi_i + b_i\sin\theta\cos\phi_i + y_{i0}\}, \end{aligned} \tag{4.2}$$

where $\theta \in [0, 2\pi]$. The expression the points on the ellipse is used to calculate the collision distances and the associated rotation angles for multiple elliptical agents in the formation control problem addressed in this study.

4.2.2 Agent dynamics

This chapter focuses on moving a group of elliptical agents to the desired formation. Consider that there are N elliptical agents in the group. The set of the agents is given as

$$E = \{E_1 \quad E_2 \quad E_3 \dots E_N\}.$$

The emphasis on the formation strategy is to overcome the difficulties caused by the elliptical shape of the agents. Therefore, the dynamics of each elliptical agent E_i is given as single integrator model

$$\begin{bmatrix} \dot{p}_i^c \\ \dot{\phi}_i \end{bmatrix} = \begin{bmatrix} u_i \\ u_{\phi_i} \end{bmatrix}. \tag{4.3}$$

For all $E_i \in E$, control variable $u_i = [u_{xi}, u_{yi}]^T$ is the position control input, and control variable u_{ϕ_i} is the heading angle control input for E_i. Vector $p_i^c = [x_{i0}, y_{i0}]^T$ represents the center position of E_i, and ϕ_i is the heading angle of E_i.

4.2.3 Formation objective

To achieve the predefined formation, the following two assumptions are imposed.

Assumption 6. *Agents can move in any 2D directions.*

Note that, by Assumption 1, there is no limitation of the moving direction of an agent. Agents can reach their predefined formation with any control laws based on the event-triggered control strategy. In addition the heading angle adaptive scheme provides the rotation control inputs for the system to lengthen the collision distance among agents to avoid collisions in motion.

Assumption 7. *For any point $p_i \in P_i$ on agent E_i, and for any point $p_j \in P_j$ on agent E_j, at the initial time t_0, the distance between these two points satisfies*

$$|p_i - p_j| > \varepsilon, \tag{4.4}$$

where $\varepsilon > 0$.

For any $E_i \in E, E_j \in E$, the distance between the final positions of elliptical agents E_i and E_j satisfies

$$|f_i^{F^s} - f_j^{F^s}| > b_i + b_j + \sigma,$$

where $\sigma > 0$. Parameters b_i and b_j are the short axis of E_i and E_j, respectively.

Notice that Assumption 7 implies that the initial position of each agent must be such that the distances among elliptical agents are sufficiently large to avoid collision in initial status. Assumption 7 is also proposed to make sure that agents in the group will not cover the final locations where other agents want to reach.

Formation objective. Under Assumptions 6 and 7, a suitable control scheme should be the one capable of driving a group of elliptical agents to a predefined formation without collisions with the other agents in the group. The predefined formation is given as

$$F = \{f_1 \quad f_2 \quad f_3 \dots f_N\}, \tag{4.5}$$

where, $f_i, i \in N$ is the desired position of E_i. The corresponding mapping for the group based on F is written as

$$F^s = \{f_1^{F^s} \quad f_2^{F^s} \quad f_3^{F^s} \dots f_N^{F^s}\}, \tag{4.6}$$

where $f_i^{F^s}$ represents the corresponding position based on f_i of E_i and the mapping relationship of the whole group. For each agent, we design u_i such that p_i^c can move to the corresponding position $f_i^{F^s}$. The design condition for u_i is such that

$$\lim_{t \to \infty} (p_i^c - f_i^{F^s}) = 0, |p_i - p_j| > \varepsilon, \tag{4.7}$$

for all $i, j \in N, i \neq j$.

4.3 Event-triggered adaptive formation control design

In event-triggered control, the control actuation times are determined by an event-triggering mechanism. To introduce the event-triggered control scheme, it is assumed that the triggering time sequence is denoted by $t^i = t_0^i, t_1^i, t_2^i, \dots t_k^i, \dots$. Agent E_i can receive the state information of the other agents in the group and update its control law u_i at $t_k^i \in t^i$. At time t_k^i, agent E_i obtains the status of the other agents, which contain the position, velocity and heading angle of each agent. The updated formation is delivered to the whole group. Based on the mapping decision and the state information, E_i can update u_i to achieve the desired formation. The event triggering scheme is given first in the following section.

4.3.1 Event-triggered scheme

The event-triggered scheme in this chapter take into consideration collision and deviation problems of each agent. Each agent has its own event-triggered strategy with its collision time and deviation time. First, the collision time for each agent is calculated. The collision set for agent E_i at t_k^i is given as

$$C_i(t_k^i) = \{E_1^{C_i}(t_k^i) \quad E_2^{C_i}(t_k^i) \quad \dots\}. \tag{4.8}$$

where $E_j^{C_i}(t_k^i)$ represents the possible collision agent in the system, which satisfies $0 < \varphi_{ij} < 180°$. Here, φ_{ij} is the angle of u_i and u_j at t_k^i, and it can be obtained by $\varphi_{ij} = \arccos(\frac{u_i(t_k^i).u_j(t_k^i)}{|u_i(t_k^i)||u_j(t_k^i)|})$. Since the system is modelled as the single-integrator system, the control input $u_i(t_k^i)$ can be seen as the velocity of agent E_i at $t \in [t_{k-1}^i, t_k^i)$, which is $u_i(t_k^i) = v_i(t_k^i)$. Also, $u_j(t_k^i)$ is the velocity of E_j at $t \in [t_{k-1}^i, t_k^i)$. The minimum distance between E_i and E_j on the direction of speed difference $\Delta v_{ij}(t_k^i) = (x_{ij}^v(t_k^i), y_{ij}^v(t_k^i))$ of these two agents is represented as $d_{ij}(t_k^i)$, while $\Delta v_{ij}(t_k^i) = u_j(t_k^i) - u_i(t_k^i)$. Here, $d_{ij}(t_k^i) = |p_i(t_k^i) - p_j(t_k^i)|$, where $p_i(t_k^i)$ and $p_j(t_k^i)$ are the corresponding points of $d_{ij}(t_k^i)$ on E_i and E_j, respectively. The coordinates of $p_i(t_k^i)$ and $p_j(t_k^i)$ are $(x_i(t_k^i), y_i(t_k^i))^T, (x_j(t_k^i), y_j(t_k^i))^T$, respectively. These two points $p_i(t_k^i)$ and $p_j(t_k^i)$ can be obtained by solving the following equations

$$\begin{cases} m_{ij}(t_k^i) = \frac{y_{ij}^v(t_k^i)}{x_{ij}^v(t_k^i)}, \\ m_{ij}(t_k^i) = \frac{y_j(t_k^i) - y_i(t_k^i)}{x_j(t_k^i) - x_i(t_k^i)}, \\ [x_i(t_k^i), y_i(t_k^i), 1] A_i(t_k^i) [x_i(t_k^i), x_i(t_k^i), 1]^T = 0, \\ [x_j(t_k^i), y_j(t_k^i), 1] A_j(t_k^i) [y_j(t_k^i), y_j(t_k^i), 1]^T = 0, \\ \psi_{ij}(t_k^i) = \arctan\left(-\frac{b_i^2\left(x_i(t_k^i) - x_{i0}(t_k^i)\right)}{a_i^2\left(y_i(t_k^i) - y_{i0}(t_k^i)\right)}\right) + \phi_i(t_k^i), \\ \psi_{ji}(t_k^i) = \arctan\left(-\frac{b_j^2\left(x_j(t_k^i) - x_{j0}(t_k^i)\right)}{a_j^2\left(y_j(t_k^i) - y_{j0}(t_k^i)\right)}\right) + \phi_j(t_k^i), \\ \psi_{ij}(t_k^i) = \psi_{ji}(t_k^i). \end{cases} \tag{4.9}$$

In (4.9), $m_{ij}(t_k^i)$ is the slope of $\Delta v_{ij}(t_k^i)$ in time interval $t \in [t_k^i, t_{k+1}^i)$, while $x_{ij}^v(t_k^i)$ and $y_{ij}^v(t_k^i)$ are the coordinate of $\Delta v_{ij}(t_k^i)$. The coordinates of points $p_i(t_k^i)$ and $p_j(t_k^i)$ are given as $p_i(t_k^i) = (x_i(t_k^i), y_i(t_k^i))$ and $p_j(t_k^i) = (x_j(t_k^i), y_j(t_k^i))$, respectively, while A_i and A_j are the coefficient matrixes of E_i and E_j, which are linked to their heading angles $\phi_i(t_k^i)$ and $\phi_j(t_k^i)$. The angle between the tangent in $p_i(t_k^i)$ and x-axis is written as $\psi_{ij}(t_k^i)$, while the angle between the tangent in $p_j(t_k^i)$ and x-axis is written as $\psi_{ji}(t_k^i)$. The two-point line based on $p_i(t_k^i)$ and $p_j(t_k^i)$ is parallel to $\Delta v_{ij}(t_k^i)$. Points $p_i(t_k^i)$ and $p_j(t_k^i)$ also satisfy (4.1). Points $p_i(t_k^i)$ and $p_j(t_k^i)$ should ensure that the tangent of the ellipse E_i at point $p_i(t_k^i)$ is parallel to the tangent of the ellipse E_j at point $x_j(t_k^i)$. Though the solution of (4.9), the corresponding points $p_i(t_k^i)$ and $p_j(t_k^i)$ on E_i and E_j can be calculated. The collision distance between agent E_i and E_j is

$$d_{ij}^{col}(t_k^i) = |p_i(t_k^i) - p_j(t_k^i)|, \tag{4.10}$$

and the collision time is

$$t_{ij}^{col}(t_k^i) = \frac{d_{ij}^{col}(t_k^i)}{|\Delta v_{ij}(t_k^i)|}. \tag{4.11}$$

Based on (4.8), (4.9), (4.10) and (4.11), the collision distance set and collision time set of agent E_i based on collision set $C_i(t_k^i)$ are given as

$$D_i^{col}(t_k^i) = \{d_1^{E_1^{C_i}(t_k^i)} \quad d_2^{E_2^{C_i}(t_k^i)} \quad \ldots\},$$
$$t_i^{col}(t_k^i) = \{t_1^{E_1^{C_i}(t_k^i)} \quad t_2^{E_2^{C_i}(t_k^i)} \quad \ldots\}.$$

The collision time of E_i at τ_k^i is obtained by finding the minimum time from the set $t_i^{col}(t_k^i)$

$$\tau_i^{col}(t_k^i) = \min \quad t_i^{col}(t_k^i). \tag{4.12}$$

Remark 4. *To ensure that the collisions will not occur among agents, the collision time of each agent should be reduced appropriate. For agent E_i, the collision time can be updated as $\tilde{\tau}_i^{col}(t_k^i) = \alpha\tau_i^{col}(t_k^i)$, where $0.5 < \alpha < 1$. If the value of α is too small, then $\tilde{\tau}_i^{col}(t_k^i)$ will be distortion, which cannot reflect the real collision relationship among agents.*

The collision relationship of agents E_i and $E_j \in C_i(t_k^i)$ at t_k^i is shown in Figure 1.

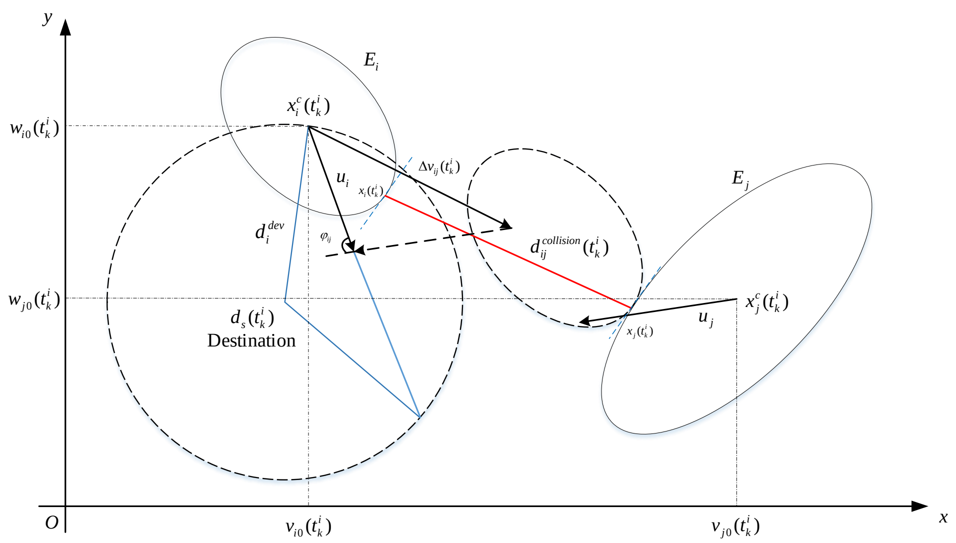

FIGURE 4.1: Collision distance and deviation distance of elliptical agent E_i at t_k^i

To consider the event-triggered scheme of each agent, the deviation time of each agent also has to be studied. The deviation time means the moving time of an agent leaving its destination if the agent moves along the current control direction. The deviation distance $d_i^{dev}(t_k^i)$ of agent E_i at t_k^i can be obtained by the corresponding points on the deviation circle of E_i. The deviation circle of E_i is shown in Figure 4.1. The corresponding point $p_i^{dev}(t_k^i)$ can be obtained by solving the following equations.

$$\begin{cases} [x_i^{dev}(t_k^i), y_i^{dev}(t_k^i), 1]B_i(t_k^i)[x_i^{dev}(t_k^i), y_i^{dev}(t_k^i), 1]^T = 0, \\ e_i(t_k^i) = \frac{u_i(t_k^i)}{|u_i(t_k^i)|}, \\ e_i(t_k^i) = \frac{p_i^{dev}(t_k^i) - p_i^c(t_k^i)}{|p_i^{dev}(t_k^i) - p_i^c(t_k^i)|}, \end{cases} \tag{4.13}$$

In (4.13), coordinates $x_i^{dev}(t_k^i)$ and $y_i^{dev}(t_k^i)$ are those of $p_i^{dev}(t_k^i)$ on the deviation circle, and $B_i(t_k^i)$, the coefficient matrix of the deviation circle, is given as followings

$$B_i(t_k^i) = \begin{bmatrix} 1 & 0 & -x_i^{F^s}(t_k^i) \\ 0 & 1 & -y_i^{F^s}(t_k^i) \\ -x_i^{F^s}(t_k^i) & -y_i^{F^s}(t_k^i) & B_1(t_k^i) \end{bmatrix}.$$

Note that the coefficient matrix $B_1(t_k^i)$ is based on the mapped desired position at t_k^i of E_i, and is given as $f_i^{F^s}(t_k^i) = (x_i^{F^s}(t_k^i), y_i^{F^s}(t_k^i))^T$, while the radius is represented by $r_i(t_k^i)$ based on the mapped desired position and the center of E_i. For $B_1(t_k^i)$, it is given as

$$B_1(t_k^i) = (x_i^{F^s}(t_k^i))^2 + (y_i^{F^s}(t_k^i))^2 - r_i(t_k^i)^2,$$

with

$$r_i(t_k^i) = |f_i^{F^s}(t_k^i) - x_i^c(t_k^i)|.$$

The unit vector of the control momentum of E_i is denoted by $e_i(t_k^i)$, while $p_i^c(t_k^i)$ is the center of E_i. The vector $\overrightarrow{(x_i^c(t_k^i), x_i^{dev}(t_k^i))}$ is parallel to $u_i(t_k^i)$. The deviation distance of E_i at t_k^i can be written as

$$d_i^{dev}(t_k^i) = |x_i^{dev}(t_k^i) - x_i^c(t_k^i)|. \tag{4.14}$$

Based on (4.14) and control input $u_i(t_k^i)$, the deviation time $t_i^{dev}(t_k^i)$ is given as

$$t_i^{dev}(t_k^i) = \frac{d_i^{dev}(t_k^i)}{|u_i(t_k^i)|}. \tag{4.15}$$

Remark 5. *To ensure a timely update for the control input, the deviation time of each agent can be appropriately reduced via the following formula*

$$t_i^{dev}(t_k^i) = \beta \frac{d_i^{dev}(t_k^i)}{|u_i(t_k^i)|}, \tag{4.16}$$

where β represents the constant coefficient. The range of β is given as $0 < \beta < 1$. Note that the smaller the β, the more frequent update the control law for the agent, leading to the deviation reduction of the desired position.

Based on the collision time and deviation time of each agent, the individual event-triggered timer at t_k^i for each agent in the system can be obtained by

$$Timer(t_k^i) = min\{\tilde{\tau}^{col}(t_k^i), t^{dev}(t_k^i)\} \leq T^{ref}, t_k^i \in t^i, \tag{4.17}$$

where $T^{ref} \in \mathbf{R}^N$ is a constant column matrix. The collision time set and deviation time set for the group of elliptical agents at t_k^i are given as $\tilde{\tau}^{col}(t_k^i) = [\tilde{\tau}_1^{col}(t_k^i)\ \tilde{\tau}_2^{col}(t_k^i)\ \ldots\ \tilde{\tau}_N^{col}(t_k^i)]^T$ and $t^{dev}(t_k^i) = [t_1^{dev}(t_k^i)\ t_2^{dev}(t_k^i)\ \ldots\ t_N^{dev}(t_k^i)]^T$, respectively. A constant matrix T^{ref} is proposed to prevent all possible collisions and an excessive deviation distance. If $Timer(t_k^i) \leq T^{ref}$, then agent E_i will update its control law, otherwise, the timer $Timer(t_k^i)$ will decrease slowly via the formula

$$Timer(t_k^i) = Timer(t_k^i) - \frac{(t^i - t_k^i)}{\Delta t} dt, t^i \in (t_k^i, t_{k+1}^i),$$

where the constant time is given as dt to decrease the *Timer* slowly when the control law is not updated, while Δt is the minimized time unit. The *Timer* of E_i will be reduced based on dt and t^i in each iteration.

4.3.2 Probability-driven controller

The probability driven controller is designed based on the event-triggered strategy. For agent E_i, the sequence of events $t^i = t_0^i, t_1^i, t_2^i, \ldots t_k^i, \ldots$ corresponds to the sequence of control updates $u_i = u_i(t_0^i), u_i(t_1^i), u_i(t_2^i), \ldots u_i(t_k^i), \ldots$. Between the event intervals, the control value will not be changed. The control input for agent $E_i \in E$ at t_k^i is given as

$$u_i(t_k^i) = \gamma R_i(t_k^i)(f_i^{F^s}(t_k^i) - x_i^c(t_k^i)), \tag{4.18}$$

where

$$R_i(t_k^i) = \begin{bmatrix} \cos\theta_i(t_k^i) & -\sin\theta_i(t_k^i) \\ \sin\theta_i(t_k^i) & \cos\theta_i(t_k^i) \end{bmatrix}.$$

In (4.18), γ is a positive constant coefficient. The rotation matrix for E_i at t_k^i is represented as $R_i(t_k^i)$, while $\theta_i(t_k^i)$ is the corresponding probability rotation angle. The value of $\theta_i(t_k^i)$ is within the interval $(-90^o, 90^o)$. The rotation matrix $R_i(t_k^i)$ is proposed to avoid the stuck problem of the agents, which will appear when two or more agents are too close to each other, and their moving directions are crossed.

The algorithm of event-triggered scheme is given in Algorithm 7.

Algorithm 7 Event-triggered scheme for the group of elliptical agents

Input: The set of corresponding mapping formation, F^s; The set of the coordinates of the centers of the elliptical agents, P^c; The set of time constant, T^{ref} The number of the elliptical agent, N;

Output: Update the control law for each agent based on event sequence.

for each $i \in N$ **do**
    Calculate the initial control input $u_i(t_0^i)$;
    Calculate the initial collision time $\tilde{\tau}_i^{col}(t_0^i)$;
    Calculate the initial deviation time $t_i^{dev}(t_0^i)$;
    $Timer_i(t_0^i) = \min(\tilde{\tau}_i^{col}(t_0^i), t_i^{dev}(t_0^i))$
    if $Timer_i(t_0^i) < T_i^{ref}$ **then**
            Agent updates its control input.
      Communicate with the others to obtain information.
    else
            Agent maintains its original control inputs.
    end if
end for
while $\sum_{i=1}^{0} |d_s(t_{k-1}^i) - p_i^c(t_{k-1}^i)| > 0$ **do**
    for each $i \in n$ **do**
        if $Timer_i \leq 0$ **then**
            Calculate the control input $u_i(t_k^i)$;
        end if
      Calculate the collision time $\tilde{\tau}_i^{col}(t_k^i)$;
      Calculate the deviation time $t_i^{dev}(t_k^i)$;
      $Timer_i(t_k^i) = \min[\tilde{\tau}_i^{col}(t_k^i), t_i^{dev}(t_k^i)]$;
        if $Timer_i(t_k^i) < T_i^{ref}$ **then**
            Agent updates its control input
            Obtain information from the other agents though communication network.
        else
                Agent maintains its original control inputs.
        end if
    end for
      $Timer(t_k^i) = Timer(t_k^i) - dt$;
end while

The following result provides a sufficient condition for reducing the distances between agents' current positions and their final desired formation.

Theorem 2. *Consider system (4.3) under Assumptions 6 and 7, if the parameter γ in (4.18) satisfies $0 < \gamma \leq \frac{-C_b+\sqrt{C_b^2-4C_aC_c}}{2p\Delta t\times C_a}$, where C_a, C_b, C_c and q are coefficients,*

$$\begin{aligned}
C_a =& 2\sum_{i=1}^{N} l_i^2(t_k^i),\\
C_b =& 2\sum_{i=1}^{N}[\cos\theta_i(t_k^i)(x_{i0}(t_k^i) - x_i^{F^s}(t_k^i))(x_i^{F^s}(t_{k+1}^i)\\
&-x_{i0}(t_k^i)) + \sin\theta_i(t_k^i)(y_{i0}(t_k^i) - y_i^{F^s}(t_k^i))(x_i^{F^s}(t_{k+1}^i)\\
&-x_{i0}(t_k^i)) + \cos\theta_i(t_k^i)(y_{i0}(t_k^i) - y_i^{F^s}(t_k^i))(y_i^{F^s}(t_{k+1}^i)\\
&-y_{i0}(t_k^i)) + \sin\theta_i(t_k^i)(y_{i0}(t_k^i)(x_i^{F^s}(t_k^i)\\
&-x_{i0}(t_k^i))(y_i^{F^s}(t_{k+1}^i) - y_{i0}(t_k^i))],\\
C_c =& \sum_{i=1}^{N}[(x_{i0}(t_k^i) - x_i^{F^s}(t_{k+1}^i))^2 + (y_{i0}(t_k^i) - y_i^{F^s}(t_{k+1}^i))^2\\
&-l_i^2(t_k^i),\\
q =& \frac{t_{k+1}^i - t_k^i}{\Delta t},
\end{aligned}$$

where l_i represents the distance between agent E_i and its desired position, $C_c < 0$, and Δt is the minimum time interval. Then the following inequality holds: $\sum_{i=1}^{N} l_i(t_k^i) \geq \sum_{i=1}^{N} l_i(t_{k+1}^i)$.

Proof. The distances between the desired position and the center of E_i at t_k^i and t_{k+1}^i are represented by $l_i(t_k^i)$ and $l_i(t_{k+1}^i)$, respectively, while $l_i(t_k^i) = |f_i^{F^s}(t_k^i) - p_i^c(t_k^i)|$ and $l_i(t_{k+1}^i) = |f_i^{F^s}(t_{k+1}^i) - p_i^c(t_{k+1}^i)|$. The desired positions of E_i at different event time are $f_i^{F^s}(t_k^i) = (x_i^{F^s}(t_k^i), y_i^{F^s}(t_k^i))^T$ and $f_i^{F^s}(t_{k+1}^i) = (x_i^{F^s}(t_{k+1}^i), y_i^{F^s}(t_{k+1}^i))^T$, respectively. The relationship with the center $p_i^c(t_{k+1}^i)$ of $E_i \in E$ at t_{k+1}^i and based on (4.18), the center $p_i^c(t_k^i)$ at t_k^i satisfies

$$p_i^c(t_{k+1}^i) = p_i^c(t_k^i) + q\Delta t \times u_i(t_k^i), \tag{4.19}$$

where $p_i^c(t_{k+1}^i) = (x_{i0}(t_{k+1}^i), y_{i0}(t_{k+1}^i))^T$, and $p_i^c(t_k^i) = (x_{i0}(t_k^i), y_{i0}(t_k^i))^T$. The minimized time unit is represented as Δt, and $q\Delta t$ is the time interval between t_k^i and t_{k+1}^i. The control input for E_i during $[t_k^i, t_{k+1}^i)$ is $u_i(t_k^i)$. Based on (4.18), (4.19) can be written as

$$p_i^c(t_{k+1}^i) = p_i^c(t_k^i) + q\Delta t \times \gamma R_i(t_k^i)(f_i^{F^s}(t_k^i) - p_i^c(t_k^i)), \tag{4.20}$$

where the constant coefficient can be given as $\eta = q\Delta t \times \gamma$

$$\eta = q\Delta t \times \gamma. \tag{4.21}$$

From (4.20), we can get the equation

$$p_i^c(t_{k+1}^i) - p_i^c(t_k^i) = \eta R_i(t_k^i) f_i^{F^s}(t_k^i) - \eta R_i(t_k^i) p_i^c(t_k^i). \tag{4.22}$$

Then we have

$$\begin{aligned}||p_i^c(t_{k+1}^i) - p_i^c(t_k^i)|| &= ||\eta R_i(t_k^i) f_i^{F^s}(t_k^i) - \eta R_i(t_k^i) p_i^c(t_k^i)|| \\ &= \eta ||R_i(t_k^i)(f_i^{F^s}(t_k^i) - p_i^c(t_k^i)|| \\ &> \eta ||f_i^{F^s}(t_k^i) - p_i^c(t_k^i)||,\end{aligned}$$

followed by

$$||p_i^c(t_{k+1}^i) - p_i^c(t_k^i)||^2 \geq \eta^2 ||f_i^{F^s}(t_k^i) - p_i^c(t_k^i)||^2. \tag{4.23}$$

Based on (4.23) and $\sum_{i=1}^N l_i(t_k^i) \geq \sum_{i=1}^N l_i(t_{k+1}^i)$, one can obtain

$$\begin{aligned}&\sum_{i=1}^N ||p_i^c(t_{k+1}^i) - p_i^c(t_k^i)||^2 \\ &\geq \eta^2 \sum_{i=1}^N ||f_i^{F^s}(t_k^i) - \eta R_i(t_k^i) p_i^c(t_k^i)||^2 \\ &\geq \eta^2 \sum_{i=1}^N l_i^2(t_{k+1}^i).\end{aligned} \tag{4.24}$$

The range of η can be calculated based on (4.24), and it can be given as

$$\eta^2 \leq \frac{\sum_{i=1}^N ||p_i^c(t_{k+1}^i) - p_i^c(t_k^i)||^2}{\sum_{i=1}^N ||f_i^{F^s}(t_k^i) - \eta R_i(t_k^i) p_i^c(t_k^i)||^2}, \tag{4.25}$$

$$1 \leq \frac{\sum_{i=1}^N l_i^2(t_{k+1}^i)}{\sum_{i=1}^N C_i}, \tag{4.26}$$

where

$$\begin{aligned}C_i =& (x_i^{F^s}(t_{k+1}^i) - x_{i0}(t_k^i))^2 + (y_i^{F^s}(t_{k+1}^i) - y_{i0}(t_k^i))^2 \\ &+ 2\eta^2 l_i^2(t_k^i) + 2\eta[\cos\theta_i(t_k^i)(x_{i0}(t_k^i) - x_i^{F^s}(t_k^i))(x_i^{F^s}(t_{k+1}^i) \\ &- x_{i0}(t_k^i)) + \sin\theta_i(t_k^i)(y_{i0}(t_k^i) - y_i^{F^s}(t_k^i))(x_i^{F^s}(t_{k+1}^i) \\ &- x_{i0}(t_k^i)) + \cos\theta_i(t_k^i)(y_{i0}(t_k^i) - y_i^{F^s}(t_k^i))(y_i^{F^s}(t_{k+1}^i) \\ &- y_{i0}(t_k^i)) + \sin\theta_i(t_k^i)(y_{i0}(t_k^i)(x_i^{F^s}(t_k^i) \\ &- x_{i0}(t_k^i))(y_i^{F^s}(t_{k+1}^i) - y_{i0}(t_k^i))].\end{aligned}$$

By solving (4.25), η can be given as

$$\frac{-C_b - \sqrt{C_b^2 - 4C_aC_c}}{2C_a} \leq \eta \leq \frac{-C_b + \sqrt{C_b^2 - 4C_aC_c}}{2C_a},$$

where

$$\begin{aligned}
C_a =& 2\sum_{i=1}^{N} l_i^2(t_k^i),\\
C_b =& 2\sum_{i=1}^{N}[\cos\theta_i(t_k^i)(x_{i0}(t_k^i) - x_i^{F^s}(t_k^i))(x_i^{F^s}(t_{k+1}^i) - x_{i0}(t_k^i))\\
&+ \sin\theta_i(t_k^i)(y_{i0}(t_k^i) - y_i^{F^s}(t_k^i))(x_i^{F^s}(t_{k+1}^i) - x_{i0}(t_k^i))\\
&+ \cos\theta_i(t_k^i)(y_{i0}(t_k^i) - y_i^{F^s}(t_k^i))(y_i^{F^s}(t_{k+1}^i) - y_{i0}(t_k^i))\\
&+ \sin\theta_i(t_k^i)(y_{i0}(t_k^i)(x_i^{F^s}(t_k^i) - x_{i0}(t_k^i))(y_i^{F^s}(t_{k+1}^i)\\
&- y_{i0}(t_k^i))],\\
C_c =& \sum_{i=1}^{N}[(x_{i0}(t_k^i) - x_i^{F^s}(t_{k+1}^i))^2 + (y_{i0}(t_k^i) - y_i^{F^s}(t_{k+1}^i))^2\\
&- l_i^2(t_k^i)].
\end{aligned}$$

It can be seen that when $\eta > 0$, and $C_c < 0$,

$$0 < \eta \leq \frac{-C_b + \sqrt{C_b^2 - 4C_aC_c}}{2C_a}. \tag{4.27}$$

Based on (4.21) and (4.27), γ satisfies the following condition

$$0 < \gamma \leq \frac{-C_b + \sqrt{C_b^2 - 4C_aC_c}}{2q\Delta t \times C_a}, \tag{4.28}$$

□

Remark 6. *As the value of γ is based on (4.28), hence the event-triggered control strategy can ensure that $\sum_{i=1}^{N} l_i(t_k^i) \geq \sum_{i=1}^{N} l_i(t_{k+1}^i)$. This means that the sum of differences between current positions and desired positions of the agents in the system will reduce or remain unchanged during the moving period, which reflects that the event-triggered formation algorithm will drive the whole group to the desired formation effectively.*

It should be mentioned that the algorithms developed in this chapter is used for discrete systems with the minimum time interval Δt, which indicates that the control momentum of each agent can only update after Δt of the time of the previous update. Thus, each agent can only update $1/\Delta t$ times in 1 second, instead of triggering countless times in a short time period, thus the Zeno-behavior will not occur in our event-triggered formation strategy.

The triggering condition for agent E_i is given as follow:

$$|p_i^c(t^i) - p_i^c(t_k^i)| \leq u_i(t_k^i)Timer(t_k^i), t^i \in [t_k^i, t_{k+1}^i), \tag{4.29}$$

where $p_i^c(t)$ is the position of E_i in t^i.

4.3.3 Adaptive strategy for elliptical agents

To enhance the performance of the event-triggered algorithm for a group of elliptical agent, the adaptive mapping algorithm and the heading angle control strategy are developed.

The proposed mapping decision for the whole group is based on the predefined formation F in (4.5). The desired position for each agent is obtained based on $L(t_k^i)$ at t_k^i, while

$$L(t_k^i) = \sum_{i=1}^{N} |f_i^{F^s}(t_k^i) - x_i^c(t_k^i)|. \tag{4.30}$$

The mapping algorithm should guarantee that the value of $L(t_k^i)$ is the minimum at t_k^i, and Algorithm 8 shown below is the mapping algorithm.

Algorithm 8 Mapping strategy for the group of elliptical agents

Input: The predefined formation, F; The set of the coordinates of the centers of the elliptical agents, X^c; The number of the elliptical agent, N;
Output: The set of mapping formation, F^s.

1: **for** each $i \in N$ **do**
2: **for** $j \in [1 : i-1, i+1 : N]$ **do**
3: Calculate the initial mapping distance $d_{ii} = |F_i - x_i^c|$;
4: Calculate the set of distance with other desired positions $d_{ij} = |F_j - x_i^c|, D_i = d_{ij}$;
5: **if** $d_{ii} < \min D_i$ **then** $f_i^{F^s} = f_i$;
6: **else** $f_i^{F^s} = f_j, d_{ij} = \min D_i$
7: **end if**
8: **end for**
9: **end for**

In addition, the heading angle control scheme is designed to prevent all possible collisions among agents. The heading angle of each elliptical agent is adjusted based on the collision distances among the agents. The angle control law is given as

$$\bar{u}_{\phi_i} = \rho \frac{\pi}{180^\circ}, \tag{4.31}$$

where $\rho = [1\ 0\ -1]^T$. Based on (31), the change angle of agent $E_i \in E$ at t_k^i, represented by $\bar{\phi}_i(t_k^i)$, is written as

$$\bar{\phi}_i(t_k^i) = \phi_i(t_k^i) + \bar{u}_{\phi_i}. \tag{4.32}$$

The collision distances of E_i from their collision set $D_i^{col}(t_k^i)$ are derived from (4.10), and the minimum collision distance of $d_j^{D_i^{col}(t_k^i)}$ is given as

$$d_j^{D_i^{col}(t_k^i)} = \min D_i^{col}(t_k^i), \qquad E_j \in C_i(t_k^i).$$

The reference collision distances between E_i and E_j are derived using (4.9), (4.10) and (4.32), which are denoted as $\bar{d}_j^1(t_k^i)$, $\bar{d}_j^2(t_k^i)$ and $\bar{d}_j^3(t_k^i)$. To avoid any potential collision, we have

$$d_j^{D_i^{col}(t_k^i)} = \max\{\bar{d}_j^1(t_k^i), \bar{d}_j^2(t_k^i), \bar{d}_j^3(t_k^i)\}. \quad (4.33)$$

The heading angle control input and final heading angle of E_i at t_k^i are written as

$$u_{\phi_i}(t_k^i) = \bar{u}_{\phi_i}(\varsigma), \quad (4.34)$$

$$\phi_i(t_k^i) = \phi_i(t_k^i) + \bar{u}_{\phi_i}(\varsigma), \quad (4.35)$$

where ς is corresponding to $d_j^{D_i^{col}(t_k^i)}$. The rotation algorithm, Algorithm 9, is given as follows.

Algorithm 9 Rotation algorithm for the group of elliptical agents

Input: The collision distance set, D^{col}; The set of the heading angles, ϕ;
Output: Update the set of the heading angles, ϕ.

for each $i \in n$ **do**
    Calculate $\bar{u}_{\phi_i}$;
    Calculate $\bar{\phi}_i$;
    for $d_j \in D_i^{col}$ **do**
        if $d_j = \min D_i^{col}$ **then**
            Calculate $\bar{d}_j^1(t_k^i)$,$\bar{d}_j^2(t_k^i)$,$\bar{d}_j^3(t_k^i)$;
            $d_j = \max\{\bar{d}_j^1(t_k^i), \bar{d}_j^2(t_k^i), \bar{d}_j^3(t_k^i)\}$
            Obtain ς corresponded to d_j.
            $u_{\phi_i} = \bar{u}_{\phi_i}(\varsigma)$
            $\phi_i = \bar{\phi}_i(\varsigma)$
        end if
    end for
end for

4.4 Simulation examples

In this section, simulation results are given to illustrate the effectiveness of the proposed event-triggered probability-driven adaptive formation control algorithm. There are ten elliptical agents in the group, and all the agents use the same control strategy. The elliptical shape of each agent differs. The parameters α in Remark 4 and β in (4.16) are given as $\alpha = 0.8$ and $\beta = 0.6$, respectively. The coefficient γ in (4.18) is valued as $\gamma = 0.5$. Each agent can receive the state and velocity information from other agents through their event sequence. The initial positions and heading angles of ten elliptical agents are given as follows.

$$\begin{aligned}
[v_{10}, w_{10}, \phi_1]^T &= [12, 13, 212^\circ]^T, \\
[v_{20}, w_{20}, \phi_2]^T &= [0, 13, 127^\circ]^T, \\
[v_{30}, w_{30}, \phi_3]^T &= [-12, 13, 341^\circ]^T, \\
[v_{40}, w_{40}, \phi_4]^T &= [-12, -13, 76^\circ]^T, \\
[v_{50}, w_{50}, \phi_5]^T &= [12, -13, 42^\circ]^T, \\
[v_{60}, w_{60}, \phi_6]^T &= [0, 6, 14^\circ]^T, \\
[v_{70}, w_{70}, \phi_7]^T &= [6, 0, 0^\circ]^T, \\
[v_{80}, w_{80}, \phi_8]^T &= [-6, 0, 25^\circ]^T, \\
[v_{90}, w_{90}, \phi_9]^T &= [0, -6, 60^\circ]^T, \\
[v_{00}, w_{00}, \phi_0]^T &= [15, -4, 150^\circ]^T.
\end{aligned}$$

The agents' elliptical properties are

$$\begin{aligned}
[a_1, b_1] &= [4, 3], [a_2, b_2] = [2, 6], \\
[a_3, b_3] &= [3, 1], [a_4, b_4] = [5, 3], \\
[a_5, b_5] &= [2, 4], [a_6, b_6] = [4, 2], \\
[a_7, b_7] &= [3, 1], [a_8, b_8] = [2, 1], \\
[a_9, b_9] &= [4, 3], [a_{10}, b_{10}] = [5, 2].
\end{aligned}$$

The predefined 2D spatial formation $F = \{f_1 \quad f_2 \quad \dots f_{10}\}$ is given as

$$\begin{aligned}
f_1 &= [10, -13], & f_2 &= [12, 16], \\
f_3 &= [-16, 19], & f_4 &= [-16, -13], \\
f_5 &= [7, 0], & f_6 &= [-8, 10], \\
f_7 &= [0, 2], & f_8 &= [-16, 0], \\
f_9 &= [0, -13], & f_{10} &= [10, 7].
\end{aligned}$$

The reference constance column matrix T^{ref} is a 10×1 matrix, in which each element is generated between 1 and 0 randomly. The reference constant column matrix T_1^{ref} for Agent 1 to Agent 5 is

$$T_1^{ref} = [0.7418\ 0.2768\ 0.8868\ 0.9505\ 0.7591]^T.$$

The reference constant column matrix T_2^{ref} for Agent 6 to Agent 10 is

$$T_2^{ref} = [0.8183\ 0.8073\ 0.5442\ 0.7416\ 0.3784]^T.$$

The simulation examples presented below aim i) to illustrate the effectiveness of event-triggered formation control algorithm in the process of achieving desired formation, each agent employs only the event-triggered scheme; and ii) to verify the enhanced performance of the proposed control strategy when the adaptive algorithm is incorporated.

Example 1 The operation of the event-triggered probability-driven control algorithm, Algorithm 7, is illustrated in this example. Figure 4.2 shows the trajectories of the ten elliptical agents reaching from their initial position and orientation to their final locations as defined by the predefined formation F. The legends $*$ and $\triangle$ used in Figure 4.2 denote, respectively, the predefined destination positions of the group and the initial position of each agent. The numbers in black color represent the locations in the predefined formation F. The misalignment between the temporary formation and desired formation is displayed in Figure 4.3. The responses as shown in Figure 4.3 approach 0 uniformly. It can be seen that there is no increase in distance while the group moves. The control inputs are shown in Figure 4.4 and Figure 4.5. We observe that the control inputs are updated only at the event sequence, and the event sequence of each agent is different based on its collision time and deviation time. The minimum collision distance is shown in Figure 4.6. The minimum distance between each agent and other agents is always positive, which indicates that there is no collision among ten elliptical agents throughout their movement.

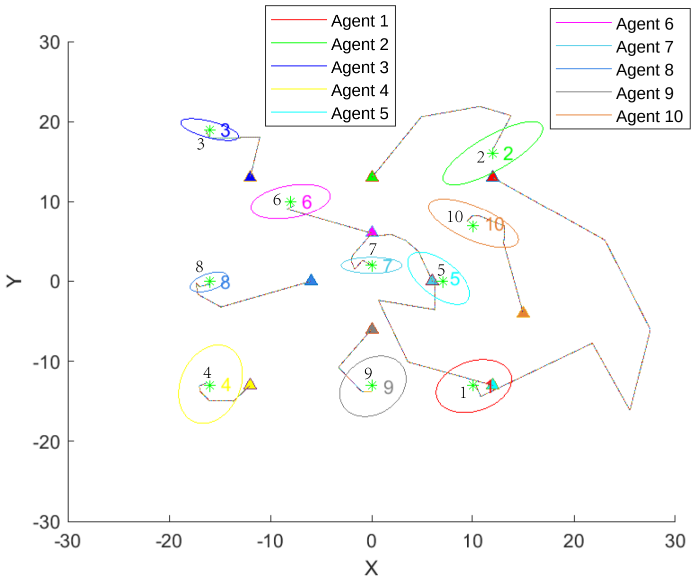

FIGURE 4.2: Trajectories of ten elliptical agents without adaptive mapping algorithm

Example 2 In this example, the mapping adaptive algorithm given in Algorithm 2 and the heading angle adaptive algorithm in Algorithm 9 are incorporated in the event-triggered control scheme of each elliptical agent. The mapping adaptive algorithm is employed to find the optimal mapping decision for each agent. Figure 4.7 shows the trajectories of ten elliptical agents, and it can be seen that agents achieve the nearest positions based on F with the minimum distance to the predefined formation of the whole group. The

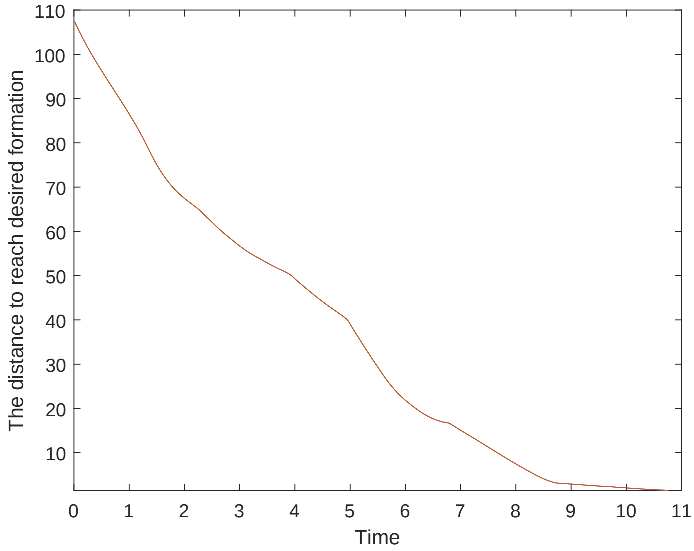

FIGURE 4.3: Distance to reach desired formation

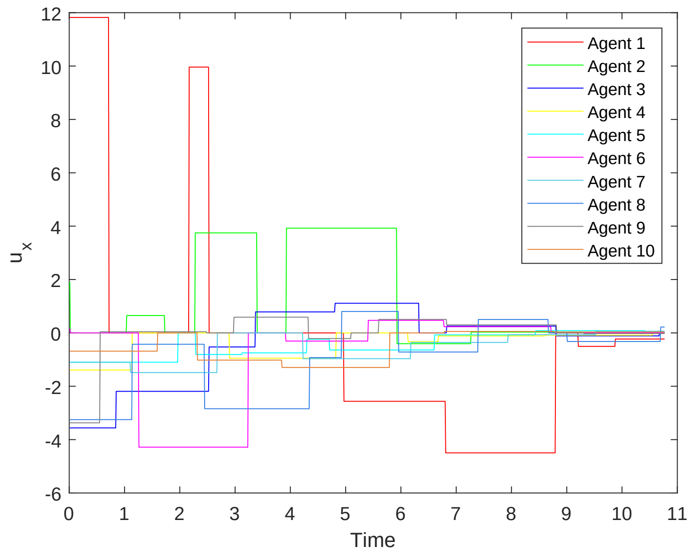

FIGURE 4.4: Control signal u_x of the elliptical agents

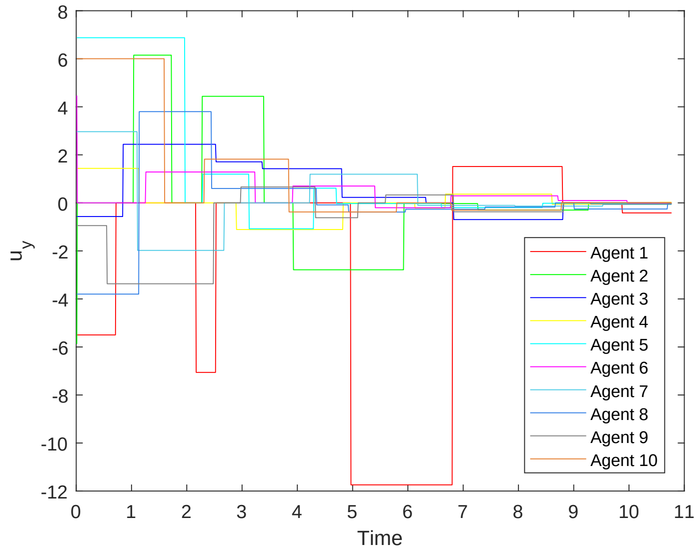

FIGURE 4.5: Control signal u_y of the elliptical agents

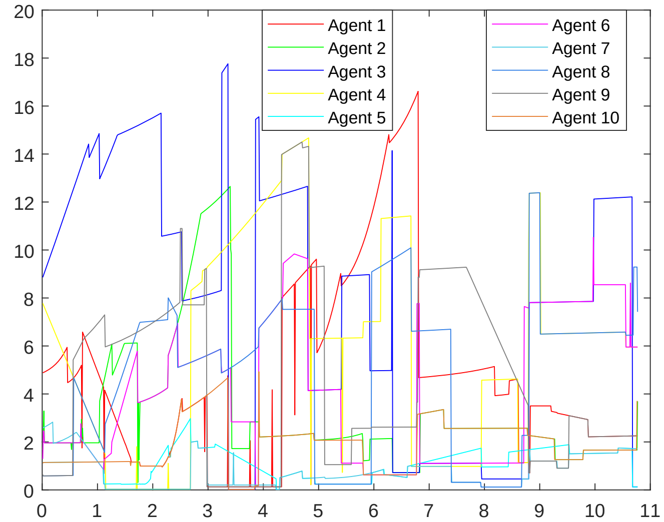

FIGURE 4.6: Changes in minimum collision distance of the elliptical agents

misalignment between the temporary formation and the predefined formation is displayed in Figure 4.8. Compared with Figure 4.3, the distance between initial positions and final formation in Figure 4.8 is about 80, which is much smaller than the initial distance in Figure 4.3, which is about 110. The control inputs for each elliptical agent are shown in Figure 4.9 and Figure 4.10. Shown in Figure 4.11 is the change of minimum collision distance of each agent. The minimum distance between each agent and other agents is greater than that in Figure 4.6. The change of heading angle based on the rotation algorithm is shown in Figure 4.12. The heading angle of each elliptical agent is changing to expand the minimum distance between each agent to further prevent collision among agents.

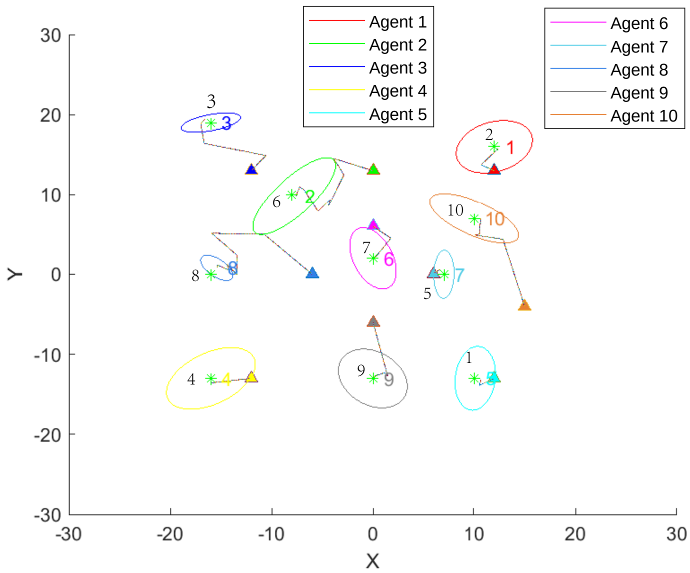

FIGURE 4.7: Trajectories of ten elliptical agents with adaptive mapping algorithm and the rotation algorithm

4.5 Conclusion

This chapter proposed an event-triggered control algorithm to drive a group of elliptical agents to a predefined formation. The control input update for each agent was event-driven, depending on the minimum collision time and deviation time of each agent. Each individual agent has its own event sequence. It can receive the state and velocity information of the others at the time when an event is triggered. The probability-driven control law is developed to prevent the stuck problem. Also, adaptive algorithms of mapping and angle rotation are proposed to enhance the performance of event-triggered control algorithm. Mapping is updated based on the minimum distance of distance to reach predefined formation. The rotation algorithm is employed to expand the minimum collision distance among agents. Simulation results of the event-triggered control algorithm and event-triggered

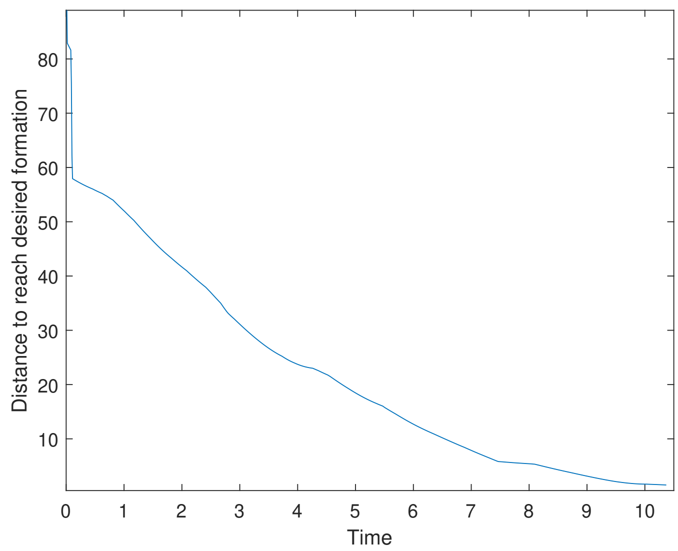

FIGURE 4.8: Distance to reach desired formation

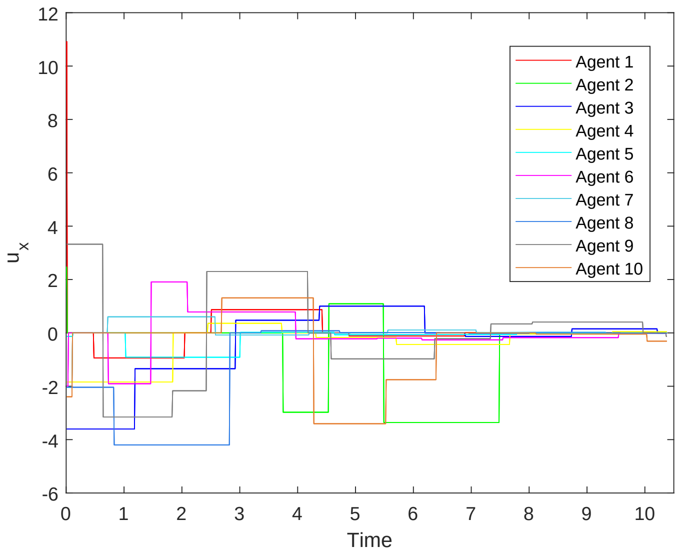

FIGURE 4.9: Control signal u_x of the elliptical agents

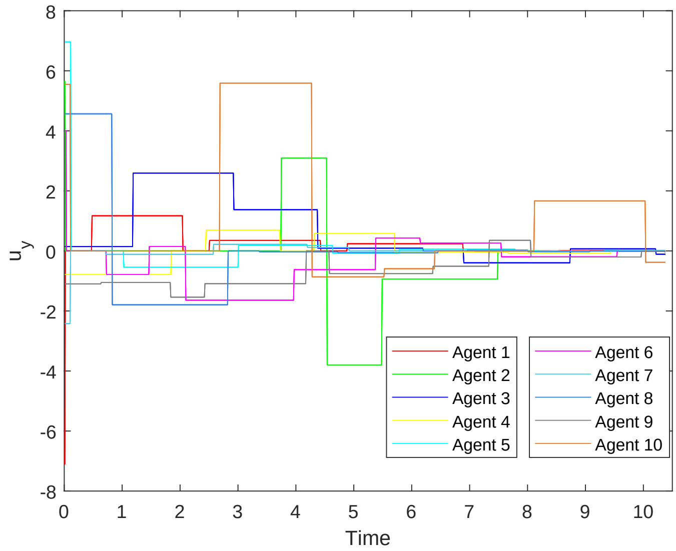

FIGURE 4.10: Control signal u_y of the elliptical agents

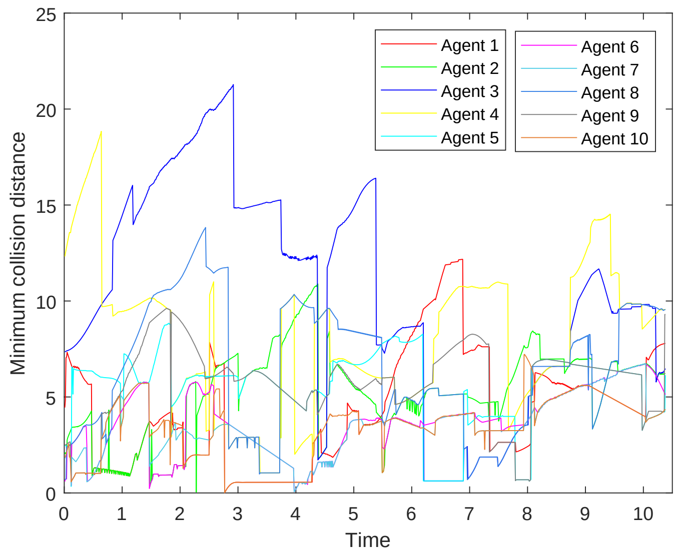

FIGURE 4.11: Changes in minimum collision distance of the elliptical agents

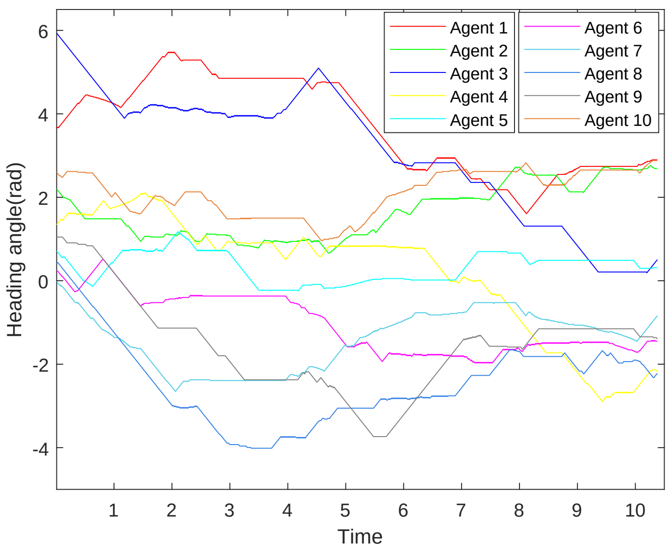

FIGURE 4.12: Changes in heading angle of the elliptical agents

adaptive control algorithm were given to demonstrate the feasibility and effectiveness of the new control design scheme.

5 Two-stage reconfiguration strategy for multi-agent systems

5.1 Introduction

In this chapter, a two-stage reconfiguration strategy is presented for a group of agents to find its special formation, which can be seen as transition of the predefined formations, during idle time in order to minimize the reconfiguration time. The basic reconfiguration strategy combines with a random mapping algorithm to find optimal special formation. To meet the practical requirements, agents are modeled as circles or ellipses. The anti-overlapping strategies are built to construct the achievable special formation based on the geometric properties of circle and ellipse. Several examples with analysis are presented to demonstrate the effectiveness and potential of the new design technique.

To enhance the usability of multi-agent systems, reconfiguration strategy should be proposed for execution of multiple formation tasks. In this chapter, a two-stage reconfiguration strategy with a random mapping algorithm is proposed to find the optimal special formation for a group of agents during the idle time. The idle time is the time interval between two tasks. This optimal special formation is found to minimize the expected reconfiguration time. Agents are formed as dots, circles and ellipses, respectively. In [144], [145], the mapping relationships between the group of agents and the predefined desired formations is fixed. The agents are all considered as points, which will not produce the overlapping problem in special designated formation. In this chapter, mapping relations between current formation of the group of agents and the predefined formations are adaptive based on the current formation of the agents. The random mapping algorithm proposed in this chapter is developed from [163]. The optimal special formation can be obtained based on the probability of the occurrence of each predefined formation, the corresponding absolute positions of each agent and the mappings achieved by the random mapping algorithm. In this chapter, the two-stage reconfiguration strategy is applied into a group of dot agents to verified the usability of this algorithm. To get closer to reality, the agents should have their own shapes, which are studied in much literature [94], [167]–[169]. In this chapter, agents are studied as dot agents, circular agents and elliptical agents. Once the agents are considered with their own shapes (circles and ellipses), the overlapping problem will occur when only introduce the two-stage reconfiguration strategy used in dot agents. Hence, the two-stage reconfiguration strategy with the random mapping algorithm should be updated based on the geometric features of the agents. The effectiveness of the two-stage reconfiguration strategy for dot agents, circular agents and elliptical agents are illustrated by simulation.

In this chapter, the main work is as follows. First, a two-stage reconfiguration strategy based on dot agents is proposed during idle time with a random mapping algorithm. Different with [144], [145], the mapping relationship applied in this chapter is changing based on the current positions of the agents and the predefined formation. This mapping algorithm is constructed based on the minimum expected moving distance between the current positions of the group of agents and each predefined formation. Second, to meet the practical requirements, the two-stage reconfiguration scheme is improved due to the circular shapes of the agents, which are used to find the optimal special formation without overlapping problem during idle time. Third, the two-stage reconfiguration strategy for elliptical agents is developed in view of geometric features of the elliptical agents to deal with the overlapping problem happens among agents.

Notation. Throughout this chapter, $\mathbf{R}^m$ is an mth dimensional space of real numbers. The symbol $|\cdot|$ represents the length of a vector. For any two points a and b, $\overrightarrow{(a,b)}$ represents the vector between a and b. Matrices are assumed to be compatible for algebraic operations. If the dimensions of matrices are not explicitly stated, they are assumed to be compatible for algebraic operations.

5.2 Two-stage reconfiguration strategy for dot agents

In this section, a two-stage reconfiguration is developed with a random mapping algorithm for dot agents. A special formation should be constructed during idle time T to simplify the movement process in next process. The position of each agent in this special formation can be obtained based on the probability of each predefined formation being the next task and the mapping relations of the current positions for all agents and each predefined formation. Idle time T is a period time between two movement process. In our two-stage reconfiguration strategy, it is assumed that idle time T is fixed and long enough to accomplish the calculation and movement to the special formation. In the N agents are assumed to form the multi-agent system,

$$E = \{E^1 \; E^2 \; \cdots \; E^N\}. \tag{5.1}$$

The predefined formation set with ϱ predefined formation is given as

$$F = \{F^1 \; F^2 \; \cdots \; F^\varrho\}, \tag{5.2}$$

while $F^s \in F$ represents the sth predefined formation which is given as

$$F^s = \{f_1^s \; f_2^s \; \cdots \; f_N^s\} \in \mathbb{R}^{2N}. \tag{5.3}$$

In (5.2), f_i^s is the position of ith point in F^s, where $f_i^s \in \mathbb{R}^2$. The current positions for the multiple agents are given as

$$P = \{p_1 \; p_2 \; \cdots \; p_N\} \in \mathbb{R}^{2N}, \tag{5.4}$$

where $p_j = (x_j^c, y_j^c)$, x_j^c and y_j^c are the x-axis coordinate and y-axis coordinate for agent E_j, respectively. The special formation which is constructed during idle time T can be written as

$$P(a) = \{p_1(a)\ p_2(a)\ \cdots\ p_N(a)\} \in \mathbb{R}^{2N}, \tag{5.5}$$

where $p_j(a) = (x_j^c(a), y_j^c(a))$, $x_j^c(a)$ and $y_j^c(a)$ are the x-axis coordinate and y-axis coordinate for agent E_j in special formation, respectively. Instead of the fixed relation between each agent and each predefined formation, the changing mappings are calculated to reduce the moving distance. The mapping relations can be obtained by the random mapping algorithm. The random mapping algorithm is based on the minimum moving distance for the whole group of agents. For the sth predefined formation F^s, to find the optimal mapping relation between F^s and P, the random mapping algorithm is described as follows.

Initialization. In the first κ iterations, η mapping relations will be generated in each iteration. In the kth iteration, the mapping relations are given as

$$M^s(k) = \{M_1^s(k)\ M_2^s(k)\ \cdots\ M_\eta^s(k)\}, 0 < k \leq \kappa, \tag{5.6}$$

$$M_i^s(k) = \{m_1^s(k)\ m_2^s(k)\ \cdots\ m_N^s(k)\}, i \in \eta, \tag{5.7}$$

where, $M_i^s(k) \in M^s(k)$. The sum of the distance between F^s and P based on each member in $M^s(k)$ is calculated by Euclidean distance, which is given as

$$L^s(k) = \{L_1^s(k)\ L_2^s(k)\ \cdots\ L_\eta^s(k)\}, \tag{5.8}$$

$$L_i^s(k) = \sum_{j=1}^{N} |f_j^s(M_i^s(k)) - p_j|. \tag{5.9}$$

The optimal mapping $m_{op}^s(k)$ satisfies that

$$L_{op}^s(k) = \min\{L^s(k), L_{op}^s(k-1)\}, \tag{5.10}$$

where $L_{op}^s(k-1)$ represents the sum distance under optimal mapping in the $(k-1)$th iteration between F^s and P. For the first iteration, $L_{op}^s(k-1)$ can be calculated based on the initial mapping relation M_0^s which is given as a condition, while

$$M_0^s = \{\bar{m}_1^s\ \bar{m}_2^s\ \cdots\ \bar{m}_N^s\}, 1 \leq s \leq \varrho,$$
$$M_0 = \{M_0^1\ M_0^2\ \cdots\ M_0^\varrho\},$$

where M_0 is the set of the initial mapping relations of the predefined formations. In the first κ iterations, the optimal mapping in each iteration will be put into the optimal set $M_{op} \in \mathbb{R}^{N\eta}$,

$$M_{op} = \{m_{op}^s(1); m_{op}^s(2); \cdots ; m_{op}^s(\kappa)\}. \tag{5.11}$$

Then. From the $(\kappa+1)$th iteration, the similarity of M_{op} should be considered first. In the $(\kappa+1)$th iteration, if no coincident element occurs in M_{op}, which means the optimal mappings in the first κ iterations are totally different. Then η new mapping relations will be generated. The optimal mapping $m^s_{op}(\kappa+1)$ in the $(\kappa+1)$th iteration could be filtered based on (5.8), (5.9) and (5.10), while the optimal set M_{op} will be changed to

$$M_{op} = \{m^s_{op}(2); m^s_{op}(3); \cdots ; m^s_{op}(\kappa+1)\}. \tag{5.12}$$

If there are overlapping elements in each row in M_{op}, the algorithm will act as follows. Assumed that r elements are the same in each row, if $r \ll n$, r can be regarded as 0. Then, the algorithm will run as if there are no duplicate elements. If r is big enough, then these overlapping elements can be seen as the fixed elements in optimal mapping $m^s_{op}(\kappa+1)$. Then, $(n-r)$ mappings with $(N-r)$ elements in each mapping should be generated, where $m^s_i(\kappa+1) \in \mathbb{R}^{(N-r)(N-r)}, 0 < i \leq n-r$. Combined these $(N-r)$ elements and r fixed elements, based on (5.8), (5.9) and (5.10), optimal mapping $m^s_{op}(\kappa+1)$ can be obtained. Then the optimal set should be updated as (5.12). If all mappings in M_{op} are the same, the final optimal mapping m^s_{op} is written as $m^s_{op}(\kappa)$.

The random mapping algorithm will iterate until the final optimal mapping is found.

To acquire the position for each agent in the special formation, the probability of occurrence of each predefined formation in the next mission should be given first. The probability of sth predefined formation F^s is given as q^s, while q^s_j represents the probability of agent E_j in the multi-agent system to move to F^s in the next mission. It can be seen that $q^s \in [0,1]$, while $q^1 + q^2 + \cdots + q^{\varrho} = 1$. Using Lemma 1 in [144], the position for each agent in special formation can be obtained by

$$p_j(a) = \sum_{i=1}^{\varrho} q^i_j f^i_j, 1 \leq j \leq N, \tag{5.13}$$

where q^i_j is the probability of occurrence of the agent E_j in predefined formation F^i. The position corresponding to E_j agent based on m^i_{op} in the ith predefined formation. In this section, the agents are considered as dots, which means no collision will occur among the agents. Hence, the positions for all agents in the group based on (5.13) are the final positions in special formation. The algorithm for the two-stage reconfiguration algorithm for a group dot agents is given as follow.

Algorithm 10 Two-stage reconfiguration for dot agents

Input:

The set of predefined formation, F;

The set of the current positions of the centers of the dot-shape agents, P;

The number of the predefined formation, ϱ;

The number of the agent, N;

The coefficient, κ;

Initial mapping relation sets, M_0;

Probability of each predefined formation, $q = \{q^1\ q^2\ \cdots\ q^{\varrho}\}$;

Output:

Find the special formation during idle time T, $P(a)$;

for $1 \leq s \leq \varrho$ **do**
  while do
    for $k = 1 : 10000000$ **do**
      if $k \leq \kappa$ **then**
        Generate η mapping relations,
        $M^s(k) = \{M_1^s(k)\ M_2^s(k)\ \cdots\ M_{\eta}^s(k)\}$;
        for $1 \leq i \leq \eta$ **do**
          Calculate the sum distance based on M_i^s;
          $L_i^s(k) = \sum_{j=1}^{N} |f_j^s(M_i^s(k)) - p_j^c|$;
        end for
        $L_{op}^s(k) = \min\{L_1^s(k)\ L_i^s(k)\ \cdots\ L_i^s(k), \quad L_{op}^s(k-1)\}$;
        Find $m_{op}^s(k)$ based on $L_{op}^s(k)$;
        $M(op)(k,:) = m_{op}^s(k)$;
      else
        Find the number of overlapping elements in $M(op)$, r;
        if $r = 0$ or $r \ll N$ **then**
          Repeat the steps such as $k \leq \kappa$;
          Update $M(op)$;
        else if $r < N$ **then**
          r overlapping elements is seen as the fixed elements in $m_{op}^s(k)$;
          Generate $N - r$ mappings with $N - r$ elements;
          Combined $N - r$ generated elements and r fixed elements to find $L_{op}^s(k)$;
          Find $m_{op}^s(k)$ based on $L_{op}^s(k)$;
          Update $M(op)$;
        else
          The final optimal mapping $m_{op}^s = m_{op}^s(\kappa)$;
          break;
        end if
      end if
    end for
  end while
end for
for $1 \leq j \leq N$ **do**
  $p_j(sp) = \sum_{i=1}^{\varrho} q_j^i f_j^i$;
end for
Obtain the special formation $P(a) = \{p_1(a)\ p_2(a)\ \cdots\ p_N(a)\}$

5.3 Two-stage reconfiguration strategy for circular agents

In this section, the two-stage reconfiguration strategy for circular agents is developed to investigate the overlapping problem. In reality, agents always have their own shapes instead of dots, which will lead to collision and overlapping problems. To solve the possible overlapping issue, the two-stage reconfiguration strategy for circular agents is proposed to find the optimal special formation $P(c) = \{p_1(c)\ p_2(c)\ \cdots\ p_N(c)\}$ for the group.

Firstly, the special formation $P(a)$ in idle time T can be calculated by (5.13). The distance between each point in $P(a)$ should be calculated to determine whether the circular agents overlap. The distance set $D^c \in \mathbb{R}^{nn}$ can be calculated as

$$\begin{aligned} d^c_{jl} &= |p_j - p_l|, j \neq l, \\ d^c_{jl} &= \infty, \qquad j = l, \end{aligned} \tag{5.14}$$

where d^c_{jl} is the distance between the positions p_j and p_l of agent E_j and agent E_l in special formation, while p_j and p_l can be obtained by (5.13). To avoid overlapping among circular agents, reference distance set D^f is given as

$$D^f = \begin{bmatrix} d^f_{11} & d^f_{12} & \cdots & d^f_{1N} \\ \vdots & \vdots & \vdots & \vdots \\ d^f_{N1} & d^f_{N2} & \cdots & d^f_{NN} \end{bmatrix},$$

$$\begin{aligned} d^f_{jl} &= r_j + r_l + \varepsilon, j \neq l, \\ d^f_{jl} &= 0, \qquad j = l, \end{aligned} \tag{5.15}$$

where r_j and r_l are the radius of circular agent E_j and E_l, respectively. Coefficient ε is a constant parameter, while d_{jl} is the reference coincidence distance for agent E_j to agent E_l. The difference between D^f and D^c is given as

$$\begin{aligned} \tilde{D} &= D^f - D^c, \\ \tilde{d}_{jl} &= d^f_{jl} - d^c_{jl}. \end{aligned} \tag{5.16}$$

Based on $\tilde{D}$, it can be determined whether there is overlap among circular agents. For agent E_j,

(1) if all $\tilde{d}_{jl} \in \tilde{D} \leq 0$, agent E_j does not have overlapping problem;

(2) if there is $\tilde{d}_{jl} \in \tilde{D} > 0$, there is overlapping problem for E_j. Assumed that α agents coincide with E_j, the overlapping agent set of E_j is given as

$$\begin{aligned} E^j(P(c)) = \{&E^j_1(P(c))E^j_2(P(c)) \\ &\cdots E^j_\alpha(P(c))\}. \end{aligned} \tag{5.17}$$

To analyze the overlapping direction, the following decision method is proposed

$$\begin{cases} S_1 = \{(x_l^c(a) - x_j^c(a))\}, \\ S_2 = \{(y_l^c(a) - y_j^c(a))\}, \end{cases} \tag{5.18}$$

where agent $E_l \in E^j(P(c))$, and $x_l^c(a)$, $x_j^c(a)$, $y_l^c(a)$ and $y_j^c(a)$ are the x-axis and y-axis coordinates of agent E_j and E_l in special formation $P(a)$, respectively. If any $s_1, s_2 \in S_1$, $s_1 \times s_2 > 0$ or any $s_3, s_4 \in S_2$, $s_3 \times s_4 > 0$, the overlapping areas of the agent E_j and all $E_l \in E^j(P(c))$ is on one side;

(3) if there is $s_1, s_2 \in S_1, s_1 \times s_2 < 0$ and $s_3, s_4 \in S_1, s_3 \times s_4 < 0$, the overlapping areas surround E_j.

Under the above conditions, the positions for circular agents in optimal special formation $P(c)$ can be updated based on the Theorem 3 as follows.

Theorem 3. *Let the circular agents be currently positioned P. Based on the special formation $P(a)$ obtained in (13), the positions for multiple circular agents in special formation $P(c)$ can be obtained as*

(1) Under condition (1), $p_j(c) = p_j(a)$;

(2) Under condition (2),

$$p_j(c) = \begin{cases} p_j(a) - (\mu(d_{jl}^f - d_{jl}^c)\cos\theta_{jl}, 0), \\ s_1 \times s_2 > 0 \\ p_j(a) - (0, \mu(d_{jl}^f - d_{jl}^c)\sin\theta_{jl}), \\ s_3 \times s_4 > 0, \end{cases}$$

$$p_l(c) = p_l(a).$$

(3) Under condition (3),

$$p_j(c) = p_j(a).$$

$$p_l(c) = p_l(a) + (d_{jl}^f - d_{jl}^c)\frac{p_l(a) - p_j(a)}{|p_l(a) - p_j(a)|},$$

where $p_j(c)$ is the position of agent E_j in $P(c)$ and $p_j(a)$ is the position of agent E_j in $P(a)$. Distance d_{jl}^c and d_{jl}^f is obtained in (5.14) and (5.15), and θ_{jl} is the angle between between $\overrightarrow{(p_j(a), p_l(a))}$ and axis x or y based on S_1 and S_2. The parameter μ is a positive coefficient, where $0 < \mu < 1$. In (2), to calculate $p_j(c)$, agent E_l is corresponding to the maximum distance on moving direction, while the positions of E_j and E_l are given as $p_j(a) = (x_j^c(a), y_j^c(a))$ and $p_l(a) = (x_l^c(a), y_l^c(a))$.

Proof. Consider condition (1) in Theorem 3, if all $\tilde{d}_{jl} \in \tilde{D} \leq 0$ for agent E_j, it can be seen that there is no overlapping problem for E_j. Hence the position $p_j(c)$ does not need to be updated based on $p_j(a)$, which is given as

$$p_j(c) = p_j(a). \tag{5.19}$$

For condition (2) in Theorem 3, if any $s_1, s_2 \in S_1$, $s_1 \times s_2 > 0$ or any $s_3, s_4 \in S_2$, $s_3 \times s_4 > 0$, then it can be seen that all agents that overlapped with agent E_j are on one side of agent E_j. Coincidence distances D^j of E_j and $E_l \in E^j(P(c))$ can be obtained as

$$\begin{aligned} D^j &= \{d_1\, d_2\, \cdots\, d_\alpha\}, \\ d_l &= d^f_{jl} - d^c_{jl}, 1 \leq l \leq \alpha, \end{aligned} \tag{5.20}$$

where d^f_{jl} and d^c_{jl} can be obtained in (5.14) and (5.15).

If $s_1 \times s_2 > 0$, then D^j will be projected onto the x-axis as

$$\begin{aligned} D^j_p &= \{d_1^{D^j_p}\, d_2^{D^j_p}\, \cdots\, d_\alpha^{D^j_p}\}, \\ d_l^{D^j_p} &= d_l \cos\theta_{jl}, 1 \leq l \leq \alpha, \end{aligned} \tag{5.21}$$

where θ_{jl} is the angle between $\overrightarrow{(p_j(a), p_l(a))}$ and x-axis,

$$\theta_{jl} = \arccos \frac{x^c_l(a) - x^c_j(a)}{|p_l(a) - p_j(a)|}, E_l \in E^j(P(c)), \tag{5.22}$$

where $p_j(a)$ and $p_l(a)$ are the positions of E_j and E_l in $P(a)$, respectively. And $x^c_j(a)$ and $x^c_l(a)$ are the coordinates of these two agents in x-axis. Based on (5.21) and (5.22), moving vector λ_j for agent E_j is written as

$$\lambda_j = |\lambda_j| e_{jl}, \tag{5.23}$$

where $|\lambda_j|$ represents the length of λ_j for E_j,

$$\begin{aligned} |\lambda_j| &= \mu \max D^j_p \\ &= \pm\mu(d^f_{jl} - d_{jl}) \cos\theta_{jl}, \end{aligned} \tag{5.24}$$

where $0 < \mu < 1$ is a constant coefficient, and E_l is corresponded to $\max D^j_p$. Symbol $\pm$ is decided by θ_{jl}. The unit vector that $\overrightarrow{(p_j(a), p_l(a))}$ projects onto the x-axis is $e_{jl} = [\pm 1, 0]$, which is decided by θ_{jl}. If all $x^c_l(a) - x^c_j(a) > 0$, all $E_l \in E_j(P(c))$ are on the right side of E_j. Thus, $-\pi/2 < \theta_{jl} < \pi/2$, and the moving direction of E_j is along $[-1, 0]$. The moving distance is given as

$$|\lambda_j| = \mu(d^f_{jl} - d_{jl}) \cos\theta_{jl}. \tag{5.25}$$

If all $x_l^c(a) - x_j^c(a) < 0$, the circular agents in $E_j(P(c))$ distribute in the left side of E_j, while $\pi/2 < \theta_{jl} < 3\pi/2$, and $e_{jl} = [1, 0]$. The moving distance for E_j is obtained by

$$|\lambda_j| = -\mu(d_{jl}^f - d_{jl}^c)\cos\theta_{jl}. \tag{5.26}$$

If $s_3 \times s_4 > 0$, the moving direction is on y-axis, the moving distance for E_j is written as

$$|\lambda_j| = \pm\mu(d_{jl}^f - d_{jl}^c)\sin\theta_{jl}, \tag{5.27}$$

where E_l corresponds to max D_p^j, and $0 < \mu < 1$. The direction vector e_{jl} is written as $e_{jl} = [0, \pm 1]$. When all $E_l \in E_j(P(c))$ satisfy $y_l^c(a) - y_j^c(a) > 0$, all E_l are above E_j. Angles θ_{jl} satisfy $0 < \theta_{jl} < \pi$, and $e_{jl} = [0, -1]$. The moving distance is obtained by

$$|\lambda_j| = \mu(d_{jl}^f - d_{jl}^c)\sin\theta_{jl}. \tag{5.28}$$

If all $y_l^c(sp) - y_j^c(sp) < 0$, all E_l are below agent E_j, and $-\pi < \theta_{jl} < 0$. Hence, $e_{jl} = [0, 1]$, and the moving distance is written as

$$|\lambda_j| = -\mu(d_{jl}^f - d_{jl}^c)\sin\theta_{jl}. \tag{5.29}$$

Based on above derivation, the position of E_j in $P(c)$ can be written as

$$p_j(c) = \begin{cases} p_j(a) - (\mu(d_{jl}^f - d_{jl}^c)\cos\theta_{jl}, 0), s_1 \times s_2 > 0 \\ p_j(a) - (0, \mu(d_{jl}^f - d_{jl}^c)\cos\theta_{jl}), s_3 \times s_4 > 0, \end{cases} \tag{5.30}$$

where E_l is corresponding to the maximum moving distance in moving direction. The position $p_l(c)$ of $E_l \in E^j(P(c))$ in $P(c)$ is same with $p_l(a)$, which is given as

$$p_l(c) = p_l(a). \tag{5.31}$$

For condition (3) in Theorem 3, if there is $s_1 \times s_2 < 0$ and $s_3 \times s_4 < 0$, which means the overlapped agents of E_j are around E_j. The position of E_j in $P(c)$ keeps the original position in $P(a)$,

$$p_j(c) = p_j(a). \tag{5.32}$$

For $E_l \in E^j(P(c))$, the moving vector λ_l can be obtained by

$$\lambda_l = |\lambda_l| e_l^j, \tag{5.33}$$

where $|\lambda_l|$ is the length of moving distance of E_l, and e_l^j is the unit vector of moving direction.

$$|\lambda_l| = d_l = d_{jl}^f - d_{jl}^c, \tag{5.34}$$

$$e_l^j = \frac{p_l(a) - p_j(a)}{|p_l(a) - p_j(a)|}, \tag{5.35}$$

where d_l is the maximum overlapping distance between E_l and E_j. The center positions of E_j and E_l are represented by p_j and p_l, respectively. Hence the positions of E_j and $E_l \in E^j(P(c))$ are given as

$$p_j(c) = p_j(a). \tag{5.36}$$

$$p_l(c) = p_l(a) + (d_{jl}^f - d_{jl}^c)\frac{p_l(a) - p_j(a)}{|p_l(a) - p_j(a)|}. \tag{5.37}$$

□

Remark 7. *Note that, moving direction for E_j in (5.30) is decided by the angle θ_{jl} of $\overrightarrow{(p_j(a), p_l(a))}$. Different values of θ_{jl} will lead to the values of $\cos\theta_{jl}$ and $\sin\theta_{jl}$ to be positive or negative. Hence the formula of moving distance should consider this problem. The algorithm will run until the system reaches non-overlapping state.*

5.4 Two-stage reconfiguration strategy for elliptical agents

In this section, the two-stage reconfiguration strategy for a group of elliptical agents is established. The agents in the group are modeled as ellipses because many practical agents have a long and narrow shape. The objective of the two-stage reconfiguration strategy for elliptical agents is to update the positions in special formation $P(a)$ obtained in Section 5.2 to find the special formation $P^c(e)$ for elliptical agents. The special formation is written as $P(e) = \{p_1(e)\ p_2(e)\ \cdots\ p_N(e)\}$. The updated reconfiguration scheme is used to avoid overlapping problem among elliptical agents. The difficulty addressed in this section is the decision method of coincident of elliptical agents and the maximum overlap distance for two elliptical agents. First, the formula of elliptical agent E_j is given as

$$E_j : [x_j, y_j, 1]A_j[x_j, y_j, 1]^T = 0 \tag{5.38}$$

where

$$A_j = \begin{bmatrix} \frac{\cos^2\phi_j}{a_j^2} + \frac{\sin^2\phi_j}{b_j^2} & \frac{\sin 2\phi_j}{a_j^2} & \frac{2A_{j1}\cos\phi_j}{a_j^2} \\ -\frac{\sin 2\phi_j}{b_j^2} & \frac{\sin^2\phi_j}{a_j^2} + \frac{\cos^2\phi_j}{b_j^2} & \frac{2A_{j1}\sin\phi_j}{a_j^2} \\ -\frac{2A_{j2}\sin\phi_i}{b_j^2} & \frac{2A_{j2}\cos\phi_j}{b_j^2} & \frac{A_{j1}^2}{a_j^2} + \frac{A_{j2}^2}{b_j^2} - 1 \end{bmatrix},$$

where A_j is the parameter matrix based on heading angle ϕ_j with

$$A_{j1} = x_j^c \cos\phi_j + y_j^c \sin\phi_j,$$
$$A_{j2} = -x_j^c \sin\phi_j + y_j^c \cos\phi_j.$$

In A_j, a_j represents the long axes of the E_j, and b_j is the short axes. The coordinate of center of agent E_j given in (5.4) is given as (x_j^c, y_j^c). The set of points on E_j is described as follow,

$$\begin{aligned} \hat{P}_j = \{\hat{p}_j = (x_j, y_j)| \\ x_j = a_j \cos\vartheta \cos\phi_j - b_j \sin\vartheta \sin\phi_j + x_j^c, \\ y_j = a_j \cos\vartheta \sin\phi_j + b_j \sin\vartheta \cos\phi_j + y_j^c\}, \end{aligned} \tag{5.39}$$

where $\vartheta \in [0, 2\pi]$. Based on the special formation $P(a)$, the following equations for E_j and E_l should be solved to make the decision of whether there is overlapped problem based on (5.38),

$$\begin{cases} [x_j(a), y_j(a), 1]A_j[x_j(a), y_j(a), 1]^T = 0, \\ [x_l(a), y_l(a), 1]A_l[x_l(a), y_l(a), 1]^T = 0. \end{cases} \tag{5.40}$$

Four conditions will occur based on (5.40). For agent E_j:

(1) if there is no real solution in (5.40) for any $E_l \in E, E_l \neq E_j$, and $\hat{p}_j \in \hat{P}_j$ satisfies

$$[\hat{p}_j, 1]A_l[\hat{p}_j, 1] > 0,$$

or $\hat{p}_l \in \hat{P}_l$ satisfies

$$[\hat{p}_l, 1]A_j[\hat{p}_l, 1] > 0,$$

then there is no intersection between E_l and E_j.

(2) If there is no real solutions in (5.40) for any $E_l \in E$, and $\hat{p}_j \in \hat{P}_j$ satisfies

$$[\hat{p}_j, 1]A_l[\hat{p}_j, 1] < 0,$$

or $\hat{p}_l \in \hat{P}_l$ satisfies

$$[\hat{p}_l, 1]A_j[\hat{p}_l, 1] < 0,$$

then E_j and E_l are inclusion relationship in $P(a)$.

If there are more than one real solution in (5.40) for $E_l \in E$, the overlapped problem is existed in $P(a)$. Assumed β agents coincide with E_j (contain inclusion relationship), the overlapping set for E_j is written as

$$\begin{aligned} E^j(P(e)) = \{E_1^j(P(e))\ E_2^j(P(e)) \\ \cdots\ E_\beta^j(P(e))\}. \end{aligned} \tag{5.41}$$

The surrounding relationships of E_j and $E_l \in E^j(P(e))$ are specified in (3) and (4).

(3) Analyzing surrounding conditions of the overlapping direction is given as

$$\begin{cases} S_3 = \{(x_l^c(a) - x_j^c(a))\}, \\ S_4 = \{(y_l^c(a) - y_j^c(a))\}, \end{cases} \tag{5.42}$$

where $E_l \in E^j(P(e))$. If any $s_1 \in S_3, s_2 \in S_3, s_1 \times s_2 > 0$ or any $s_3 \in S_4, s_4 \in S_4, s_3 \times s_4 > 0$, then the overlapping areas of the agent E_j and all $E_l \in E^j(P(e))$ are on one side.

(4) If there is $s_1 \in S_3, s_2 \in S_3, s_1 \times s_2 < 0$ and $s_3 \in S_4, s_4 \in S_4, s_3 \times s_4 < 0$, the intersections of E_j and $E_l \in E^j(P(e))$ are around E_j.

The positions for elliptical agents in optimal special formation $P(e)$ can be updated based on the Theorem 4 as follows.

Theorem 4. *Let the elliptical agents satisfied (5.38) be currently positioned in P. Based on the special formation $P(a)$ obtained in (5.13), the positions for multiple elliptical agents in special formation $P(e)$ can be obtained as*

(1) Under condition (1), $p_j(e) = p_j(a)$;

(2) Under condition (2) and (3),

$$p_j(e) = \begin{cases} p_j(a) - ((\mu d_{jl} \cos\theta_{jl} + \varepsilon), 0), \\ s_1 \times s_2 > 0 \\ p_j(a) - (0, (\mu d_{jl} \sin\theta_{jl} + \varepsilon)), \\ s_3 \times s_4 > 0, \end{cases}$$

$$p_l(e) = p_l(a).$$

(3) Under condition (2) and (4),

$$p_j(e) = p_j(a).$$

$$p_l(e) = p_l(a) + (d_{jl} + \varepsilon)e_{jl},$$

where $p_j(e)$ is the position of agent E_j in $P(e)$ and $p_j(a)$ is the position of agent E_j in $P(a)$. Distance d_{jl} is the moving distance of E_j, and θ_{jl} is the projection angle of d_{jl} to x-axis. The constant parameter μ is used to control the length of moving distance, and $0 < \mu < 1$. Coefficient ε is proposed to ensure the moving distance. In (3), e_{jl} is the moving direction for $E_l \in E^j(P(e))$.

Proof. For condition (1) in Theorem 4, there is no coincident problem of E_j. Hence, the position $p_j(e)$ can be obtained by

$$p_j(e) = p_j(a), \tag{5.43}$$

where $p_j(a)$ is the position of E_j in special formation $P(a)$, which is calculated in (5.13).

For condition (2) in Theorem 4, all intersections of E_j are on one side of E_j. Moving vector λ_j of agent E_j can be obtained by

$$\lambda_j = \mu|\lambda_j|e_{jl} + \varepsilon, \tag{5.44}$$

where $|\lambda_j|$ is the moving distance of agent E_j, and e_{jl} is the moving direction. Parameter μ is the positive coefficient, and $0 < \mu < 1$, and ε is a positive coefficient to ensure the moving length of agent E_j. Moving distance $|\lambda_j|$ can be obtained based on the conditions of overlapping distribution. If $s_1 \times s_2 > 0, s_1 \in S_3, s_2 \in S_3$, the intersections are distributed to the left or right of E_j, and $|\lambda_j|$ can be written as

$$|\lambda_j| = \max\{d_{jl}|\cos\theta_{jl}|\}, E_l \in E^j(P(a)), \tag{5.45}$$

where d_{jl} is the moving distance of E_j based on E_l, while θ_{jl} is the angle between moving direction and x-axis.

There are three situations to calculate d_{jl} and θ_{jl}. The first situation is that there are more than two real solutions of (5.40), which means two agents have more than two points of intersection. Second situation is the inscribed situation, which is given as: there are one or two points of intersection $\hat{p}_1 = (x_1, y_1)$ and $\hat{p}_2 = (x_2, y_2)$, and there are

$$\begin{aligned}
&d_f(x,y) = \sqrt{(x - x_l^c(a))^2 + (y - y_l^c(a))^2},\\
&d_f(\hat{p}_1) > d_f(\hat{p}_j),\\
&d_f(\hat{p}_2) > d_f(\hat{p}_j),\\
&\hat{p}_j \in \hat{P}_j, \hat{p}_j \neq \hat{p}_1, \hat{p}_j \neq \hat{p}_2,
\end{aligned} \tag{5.46}$$

where $\hat{p}_j$ is points on E_j which is different from $\hat{p}_1$ and $\hat{p}_2$. The third situation is the containing situation, which is given as: there is no real solutions in (5.40), however, $\hat{p}_j \in \hat{P}_j$ satisfies

$$[\hat{p}_j, 1]A_l[\hat{p}_j, 1] < 0. \tag{5.47}$$

or $\hat{p}_l \in \hat{P}_l$ satisfies

$$[\hat{p}_l, 1]A_j[\hat{p}_l, 1] < 0, \tag{5.48}$$

which means E_j and E_l have containing relationship. The overlap area of E_j and E_l is relatively large corresponding to E_j. Hence, the moving distance d_{jl} is defined as

$$d_{jl} = \max\{a_j, a_l\}, \tag{5.49}$$

where a_j and a_l are the long axis of E_j and E_l, respectively. The angle θ_{jl} is given as

$$\theta_{jl} = \arccos\frac{x_l^c(sp) - x_j^c(sp)}{|p_l(sp) - p_j(sp)|}, \tag{5.50}$$

where $p_j(a)$ and $p_l(a)$ are the locations of E_j and E_l in special formation $P(a)$ obtained in (5.13), respectively, and $x_j^c(a)$ and $x_l^c(a)$ are the x-axis coordinates of E_j and E_l.

If there are two real solutions for (5.40), or (5.40) exists one real solution (x_1, y_1), which satisfies

$$[\hat{p}_j, 1]A_l[\hat{p}_j, 1] \geq 0, \tag{5.51}$$

where $\hat{p}_j \in \hat{P}_j$. The coincidence distance d_{jl} of E_j and $E_l \in E^j(P(e))$ can be obtained by

$$\begin{aligned}
&[x_1, y_1, 1]A_j[x_1, y_1, 1]^T = 0, \\
&[x_1, y_1, 1]A_l[x_1, y_1, 1]^T = 0, \\
&[x_2, y_2, 1]A_j[x_2, y_2, 1]^T = 0, \\
&[x_2, y_2, 1]A_l[x_2, y_2, 1]^T = 0, \\
&m_{jl} = \frac{y_1 - y_2}{x_1 - x_2}, \\
&y_{m1} = m_{jl}x_{m1} + b_1, \\
&y_{m2} = m_{jl}x_{m2} + b_2, \\
&d_{jl} = \frac{|b_1 - b_2|}{\sqrt{m_{jl}^2 + 1}}
\end{aligned} \tag{5.52}$$

where (x_1, y_1) and (x_2, y_2) are the points of intersection between E_j and E_l. The slop of line between (x_1, y_1) and (x_2, y_2) is given as m_{jl}. The points of tangents based on m_{jl} can be obtained, while (x_{m1}, y_{m1}) and (x_{m2}, y_{m2}) are two tangent points which have shortest distance. Parameters b_1 and b_2 can be calculated based on m_{jl}, (x_{m1}, y_{m1}) and (x_{m2}, y_{m2}). The overlapped distance d_{jl} between E_j and E_l will be calculated based on m_{jl}, b_1 and b_2. Angle θ_{jl} is written as

$$\theta_{jl} = \arctan(-\frac{1}{m_{jl}}). \tag{5.53}$$

The moving direction of E_j is given as $e_{jl} = [\pm 1, 0]$. Based on $s_1 \times s_2 > 0, s_1 \in S_3, s_2 \in S_3$, the moving direction of E_j is along x-axis. If all $x_l^c(a) - x_j^c(a) > 0$, all intersections of E_j distribute on the right side of E_j, $e_{jl} = [-1, 0]$. If all $x_l^c(a) - x_j^c(a) < 0$, all intersections of E_j distribute on the left side of E_j, $e_{jl} = [1, 0]$.

If $s_3 \times s_4 > 0, s_3 \in S_4, s_4 \in S_4$, the intersections distribute above or below E_j, and $|\lambda_j^c|$ can be written as

$$|\lambda_j^c| = \max\{d_{jl}|\sin\theta_{jl}|\}, E_l \in E^j(P(a)), \tag{5.54}$$

where d_{jl} can be obtained in (49) and (52). The angle θ_{jl} is calculated in (5.50) and (5.53). The moving direction of E_j is given as $e_{jl} = [0, \pm 1]$. Based on

$s_3 \times s_4 > 0, s_3 \in S_4, s_4 \in S_4$, the moving direction of E_j is along y-axis. If all $y^c_l(a) - y^c_j(a) > 0$, all intersections of E_j distribute above E_j, and $e_{jl} = [0, -1]$. If all $y^c_l(a) - y^c_j(a) < 0$, all intersections of E_j distribute below E_j, and $e_{jl} = [0, 1]$. Based on above derivation, the updated position $p_j(e)$ of E_j in $P(e)$ is given as

$$p_j(e) = \begin{cases} p_j(a) - ((\mu d_{jl}\cos\theta_{jl} + \varepsilon), 0), \\ s_1 \times s_2 > 0 \\ p_j(a) - (0, (\mu d_{jl}\sin\theta_{jl} + \varepsilon)), \\ s_3 \times s_4 > 0, \end{cases} \tag{5.55}$$

where E_l is corresponding to (5.46).

In condition (2), $E_l \in E^j(P(e))$ does not change its position based on $P(a)$, which is written as

$$p_l(e) = p_l(a). \tag{5.56}$$

For condition (3) in Theorem 4, the overlapping agents $E_l \in E^j(P(e))$ distribute around E_j. In this condition, the position of E_j in $P(e)$ keeps the original position in $P(a)$,

$$p_j(e) = p_j(a), \tag{5.57}$$

which means $|\lambda_j| = 0$. For $E_l \in E^j(P(e))$, the moving vector λ_l is calculated as

$$\lambda_l = \mu|\lambda_l|e_l + \varepsilon, \tag{5.58}$$

where μ and ε are the positive coefficients, while $|\lambda_l|$ is the length of moving distance for E_l which can be obtained based on (5.49) and (5.52). Moving direction e_{jl} of E_l is obtained based on the overlapping situations. If $|\lambda_l|$ is obtained by (5.49), moving direction e_{jl} is calculated by

$$e_{jl} = \frac{p_l(a) - p_j(a)}{|p_l(a) - p_j(a)|}, \tag{5.59}$$

where $p_l(a)$ and $p_j(a)$ are the coordinations of center of E_l and E_j in special formation $P(a)$. If $|\lambda_l|$ is obtained by (5.52), moving direction e_{jl} is calculated by

$$\begin{aligned} e_{jl} &= (\cos\theta_{jl}, \sin\theta_{jl}) \\ &= (\cos(\arctan(-\frac{1}{m_{jl}})), \sin(\arctan(-\frac{1}{m_{jl}}))). \end{aligned} \tag{5.60}$$

Hence, the updated positions of $E_l \in E^j(P(e)$ are given as

$$\begin{aligned} p_l(e) &= p_l(a) + \lambda_l \\ &= p_l(a) + (d_{jl} + \varepsilon)e_{jl}. \end{aligned} \tag{5.61}$$

□

Remark 8. *Note that the iterative moving among elliptical agents may lead to the new overlapping problem. Hence, the algorithm will loop until the special formation $P(e)$ is found without overlapping.*

5.5 Simulation results

In this section, simulation results are given to illustrate the feasibility of the two-stage reconfiguration strategy for a group of agents. Five agents form the multi-agent system, and it is assumed that all agents have the same control strategy. Three examples are provided in the simulation study to demonstrate the effect on constructing the special formation and avoiding overlapping problem with the two-stage reconfiguration strategy.

Example 1 In this example, agents are modeled as dots. Four predefined formations are given as the predefined formations, and make up the predefined formation set F, which are given below

$$\begin{aligned}
&F^1 = \{(5\ \ 8),(7\ \ 8),(9\ \ 8),(6\ \ 6),(8\ \ 6)\},\\
&F^2 = \{(-9\ \ -12),(-7\ \ -15),(-5\ \ -18),\\
&\quad (-3\ \ -15),(-1\ \ -12)\},\\
&F^3 = \{(-15\ \ 5),(-12\ \ 3),(-15\ \ 1),(-9\ \ 1),(-9\ \ 5)\},\\
&F^4 = \{(11\ \ -3),(12\ \ -5),(14\ \ -4),(13\ \ -7),(15\ \ -6)\}.
\end{aligned}$$

The probabilities of the predefined formations F^1, F^2, F^3 and F^4 are given by $q^1 = 0.15$, $q^2 = 0.3$, $q^3 = 0.35$, and $q^4 = 0.2$, respectively, and $q^1 + q^2 + q^3 + q^4 = 1$. The different special formation generated by Algorithm 10 based on different P is given in Figures 5.1-5.4. In these figures, legend $*$ denotes the coordinates of the points in the predefined formations, legend $\triangle$ represents the current positions P for these agents, and legend $\circ$ represents the special formation for the system. It can be seen that the special formations have different forms because the current position of each agent leads the different mapping relationships to the possible formations. The corresponding mapping relationships $m_{op}(F)$ to these four predefined formations with $\kappa = 10$ and $\eta = 5$ are given below.

$$m_{op}(F^1) = \left\{\begin{array}{ccccc} 1 & 2 & 3 & 4 & 5\\ 1 & 2 & 5 & 4 & 3\\ 5 & 1 & 2 & 3 & 4\\ 1 & 4 & 3 & 2 & 5 \end{array}\right\}.$$

$$m_{op}(F^2) = \left\{\begin{array}{ccccc} 1 & 4 & 2 & 3 & 5\\ 1 & 2 & 3 & 4 & 5\\ 3 & 1 & 2 & 4 & 5\\ 1 & 3 & 4 & 5 & 2 \end{array}\right\}.$$

$$m_{op}(F^3) = \left\{\begin{array}{ccccc} 1 & 3 & 4 & 5 & 2\\ 1 & 3 & 2 & 5 & 4\\ 1 & 2 & 3 & 4 & 5\\ 3 & 2 & 4 & 5 & 1 \end{array}\right\}.$$

$$m_{op}(F^4) = \left\{\begin{array}{ccccc} 1 & 4 & 3 & 2 & 5\\ 1 & 5 & 4 & 3 & 2\\ 1 & 2 & 5 & 3 & 4\\ 1 & 2 & 3 & 4 & 5 \end{array}\right\}.$$

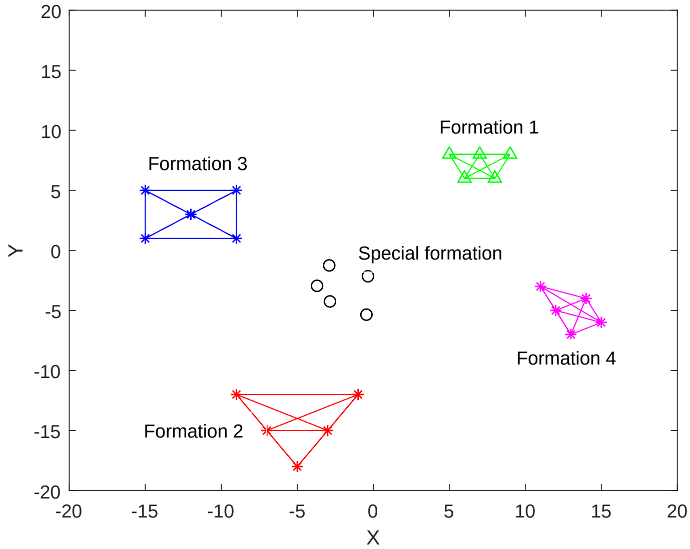

FIGURE 5.1: Special formation for five agents located in F^1

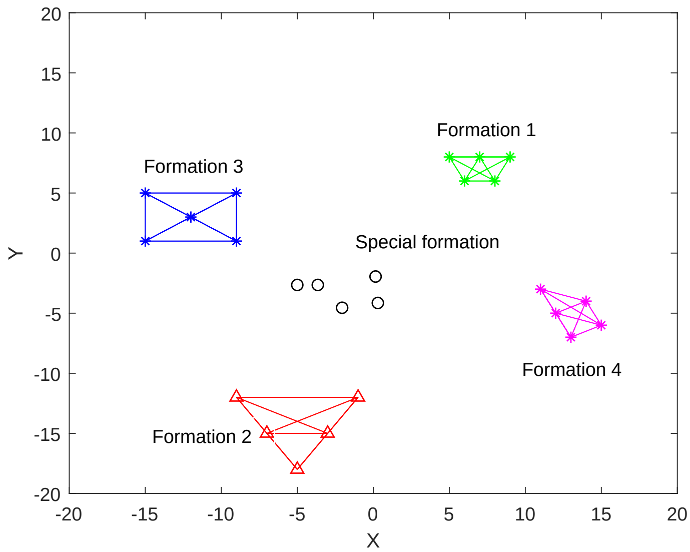

FIGURE 5.2: Special formation for five agents located in F^2

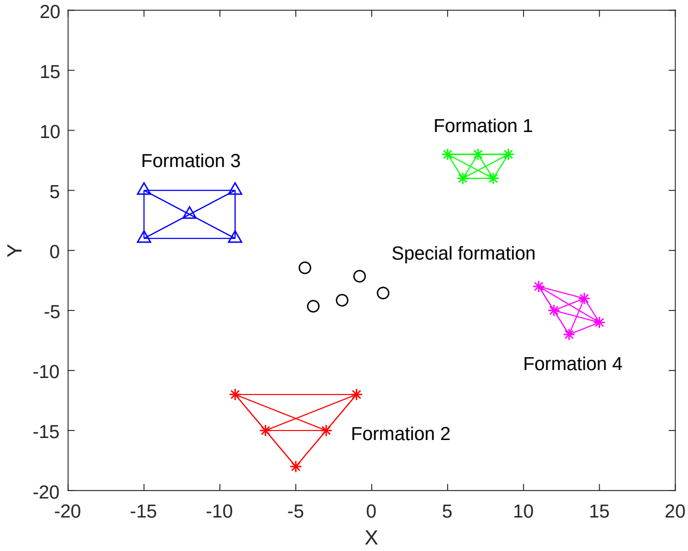

FIGURE 5.3: Special formation for five agents located in F^3

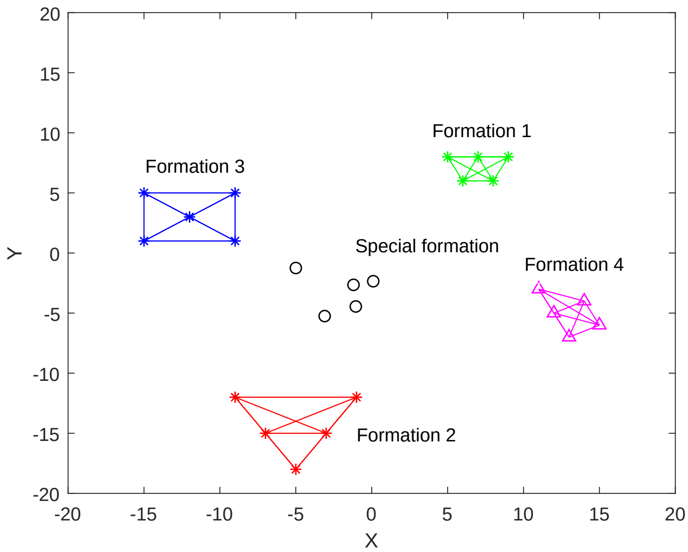

FIGURE 5.4: Special formation for five agents located in F^4

It can be seen that the different initial positions of five agents will lead the different special formations. This is because of the random mapping algorithm employed in the two-stage reconfiguration strategy.

Example 2 The agents are modeled as circle shapes. Four predefined formations are given to make up the predefined formation set F, which are given below

$$\begin{aligned}
F^1 &= \{(20\ \ 16), (28\ \ 16), (36\ \ 16), (24\ \ 8), (31\ \ 8)\},\\
F^2 &= \{(-18\ -32), (-12\ -40), (-5\ -48), (-4\ -40),\\
&\quad (-1\ -32)\},\\
F^3 &= \{(-37\ \ 15), (-33\ \ 5), (-38\ \ -2), (-29\ \ -2),\\
&\quad (-29\ \ 15)\},\\
F^4 &= \{(21\ \ -18), (27\ \ -23), (30\ \ -12), (34\ \ -25),\\
&\quad (23\ \ -31)\}.
\end{aligned}$$

Circular agents in F are not overlapping with the other agents. The radius of each agent is given as

$$r = \{2\ \ 3\ \ 4\ \ 3\ \ 2\}.$$

The parameters ε and μ are set as $\varepsilon = 0.2$, $\mu = 0.3$. The special formation for circular agents is illustrated in Figure 5.5. Circular agents are located in F^1. The final positions of the whole group are given in $P_1(c)$, which is shown as

$$\begin{aligned}
P_1(c) = \{&(-14.68\ \ -4.90),\\
&(-5.29\ \ -6.20),\\
&(4.32\ \ -6.20),\\
&(-3.20\ \ -14.10),\\
&(-5.61\ \ -23.70)\}.
\end{aligned}$$

The special formation for circular agents located in F^2 is illustrated in Figure 5.6. The final positions of the whole group are given in $P_2(c)$, which is shown as

$$\begin{aligned}
P_2(c) = \{&(-3.15\ \ -9.35),\\
&(-0.49\ \ -14.60),\\
&(-21.18\ \ -15.50),\\
&(-12.16\ \ -15.60),\\
&(0.95\ \ -4.35)\}.
\end{aligned}$$

The special formation for circular agents located in F^3 is illustrated in Figure 5.7. The final positions of the whole group are given in $P_3(c)$, which is shown as

$$\begin{aligned}
P_3(c) = \{&(-9.35\ \ -7.95),\\
&(2.62\ \ -15.25),\\
&(-12.74\ \ -15.30),\\
&(-3.72\ \ -16.56),\\
&(-0.25\ \ -4.35)\}.
\end{aligned}$$

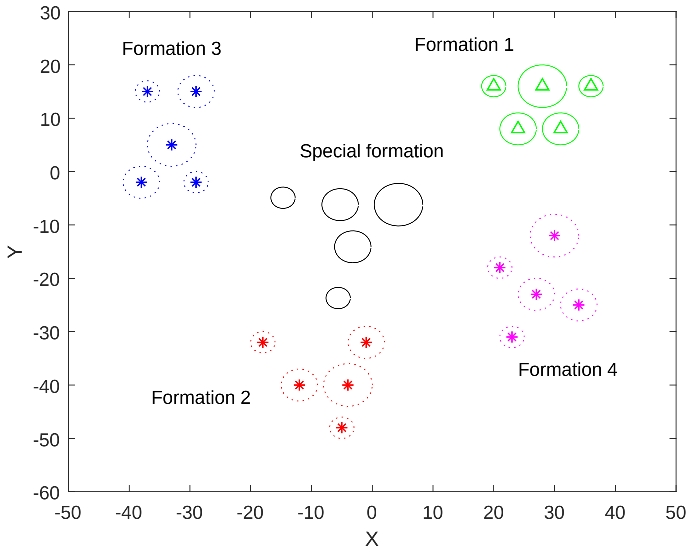

FIGURE 5.5: Special formation for five circular agents located in F^1

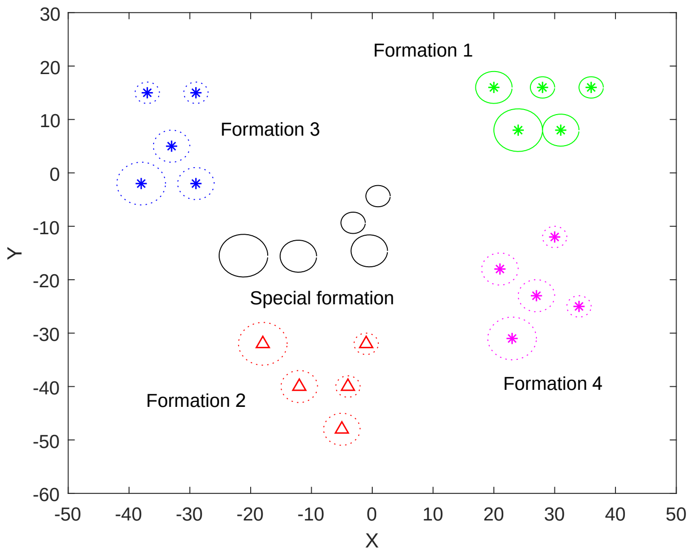

FIGURE 5.6: Special formation for five circular agents located in F^2

The special formation for circular agents located in F^4 is illustrated in Figure

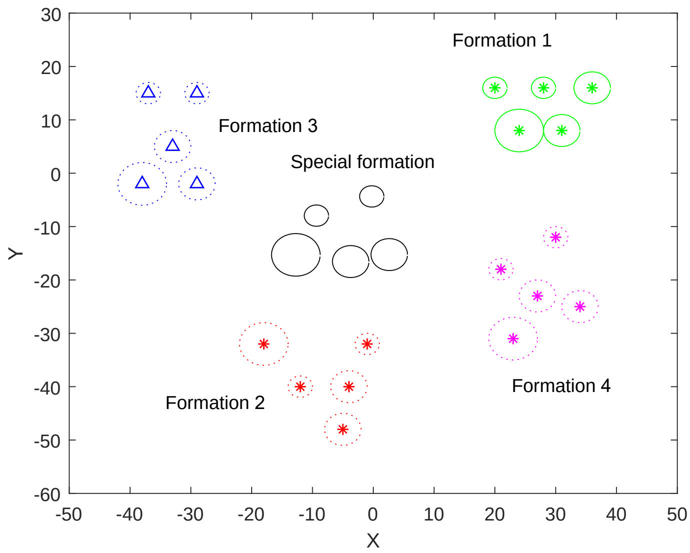

FIGURE 5.7: Special formation for five circular agents located in F^3

5.8. The final positions of the whole group are given in $P_4(c)$, which is shown as

$$\begin{aligned} P_4(c) = \{ & (-7.57 \quad -7.95), \\ & (-4.18 \quad -12.45), \\ & (-11.23 \quad -15.30), \\ & (-1.25 \quad -18.90), \\ & (0.95 \quad -4.35)\}. \end{aligned}$$

It can be seen that the special formation for these circular agents is constructed without the overlapping problem.

Example 3 In this example, the agents are modeled as ellipses. Four predefined formations are given to form the predefined formation set F, which are same with predefined formation set in Example 2. The elliptical properties of the elliptical agents are

$$\begin{aligned} & [a_1, b_1, \theta_1] = [2, 1, 0^\circ], \quad [a_2, b_2, \theta_2] = [4, 2, 127^\circ], \\ & [a_3, b_3, \theta_3] = [3, 1, 341^\circ], [a_4, b_4, \theta_4] = [5, 3, 76^\circ], \\ & [a_5, b_5, \theta_5] = [2, 1, 42^\circ]. \end{aligned}$$

Parameters of ε and μ are taken as $\varepsilon = 0.2$, $\mu = 0.3$. The special formation for elliptical agents is illustrated in Figure 5.9. The agents are located in F^1. The

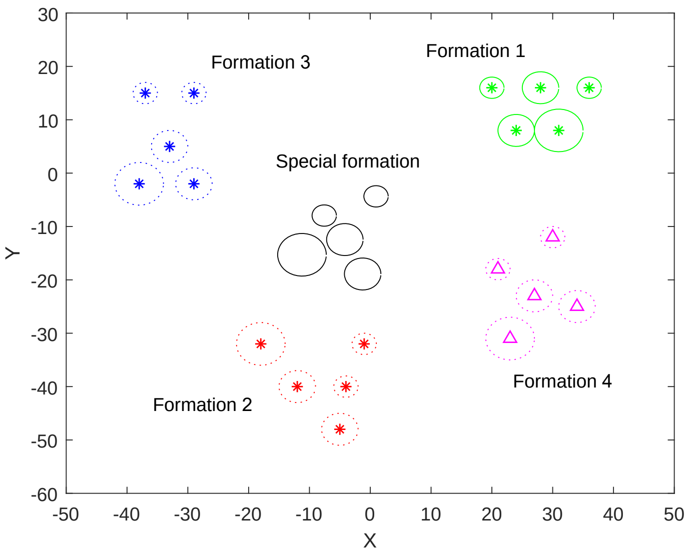

FIGURE 5.8: Special formation for five circular agents located in F^4

final position $P_1(e)$ of each agent is obtained as

$$\begin{aligned} P_1(e) = \{&(-11.15 \quad -5.55), \\ &(-3.55 \quad -6.75), \\ &(0.99 \quad -12.85), \\ &(-7.00 \quad -15.30), \\ &(-1.60 \quad -18.50)\}. \end{aligned}$$

The special formation for elliptical agents is illustrated in Figure 5.10. The agents are located in F^2. The final position $P_2(e)$ of each agent is obtained as

$$\begin{aligned} P_2(e) = \{&(-10.90 \quad -10.30), \\ &(-6.55 \quad -14.05), \\ &(-2.40 \quad -20.10), \\ &(-1.25 \quad -6.55), \\ &(-6.15 \quad -9.15)\}. \end{aligned}$$

The special formation for elliptical agents is illustrated in Figure 5.11. The agents are located in F^3. The final position $P_3(e)$ of each agent is obtained as

$$\begin{aligned} P_3(e) = \{&(-6.95 \quad -7.95), \\ &(-5.35 \quad -12.85), \\ &(-9.45 \quad -15.30), \\ &(-0.25 \quad -18.50), \\ &(-0.25 \quad -4.35)\}. \end{aligned}$$

The special formation for elliptical agents is illustrated in Figure 5.12. The

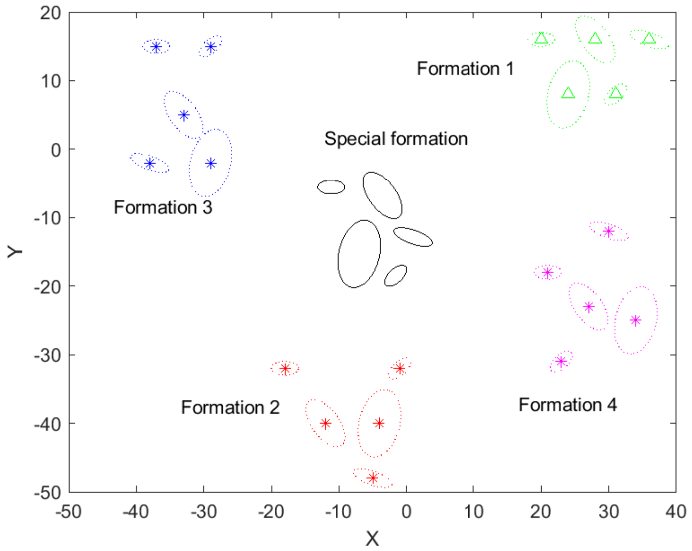

FIGURE 5.9: Special formation for five elliptical agents located in F^1

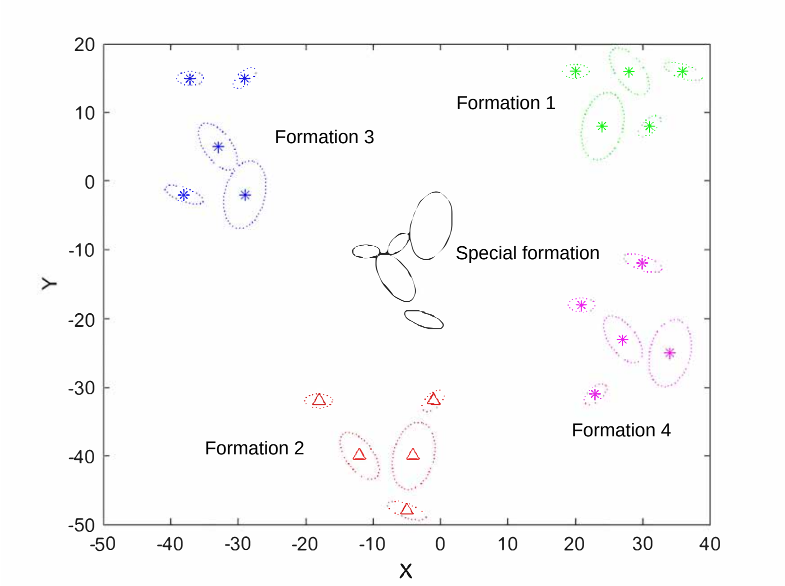

FIGURE 5.10: Special formation for five elliptical agents located in F^2

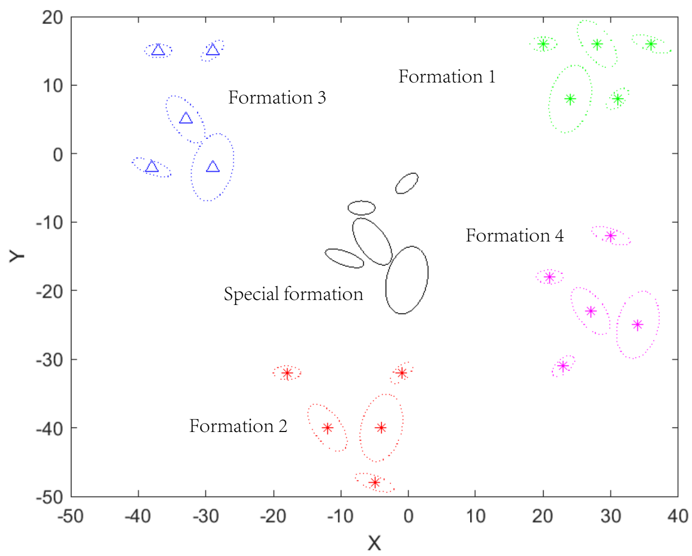

FIGURE 5.11: Special formation for five elliptical agents located in F^3

agents are located in F^4. The final position $P_4(e)$ of each agent is obtained as

$$\begin{aligned} P_4(e) = \{&(-9.95 \quad -9.05), \\ &(-5.35 \quad -9.35), \\ &(0.20 \quad -5.55), \\ &(0.85 \quad -17.30), \\ &(-6.30 \quad -17.70)\}. \end{aligned}$$

It can be seen that the five agents can find their special formation during idle time by using the two-stage reconfiguration strategy. The special formation is changing based on the mapping relationship between the current positions of these agents and each predefined formation. At the same time, circular agents and elliptical agents can avoid overlapping problems effectively.

5.6 Conclusion

This section proposed a two-stage reconfiguration strategy for a group of agents. By applying the two-stage reconfiguration algorithm during idle time, it can shorten the expected reconfiguration time when the next command with formation changing is given. These agents are modeled as dots, circles and ellipses to gradually approach the practical application. In this chapter, the two-stage reconfiguration strategy combined with the random mapping algorithm is proposed to find the special formation during idle time based on optimal mappings to predefined formation set. The two-stage reconfiguration scheme is improved for circular agents and elliptical agents to deal with the overlapping problem which may appear in the special formation by using the two-stage reconfiguration strategy for dot agents. The

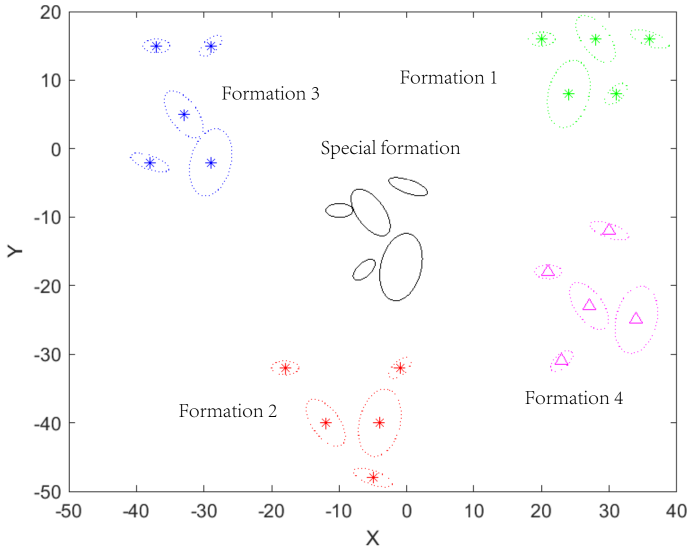

FIGURE 5.12: Special formation for five elliptical agents located in F^4

simulations of the two-stage reconfiguration strategy were given to demonstrate the feasibility and effectiveness of the new reconfiguration strategy. In our future work, the varying-probability of each predefined formation will be studied to make the reconfiguration scheme more flexible.

6 Conclusion

Collision-free formation control strategies for multiple elliptical agents and the two-stage reconfiguration strategies for multi-agent systems are considered in this thesis. The algorithms feature localised and decentralised structure and distributed computing. In Chapter 2, the formation control strategy with random mapping algorithm is proposed. The communication among elliptical agents are limited, in which only identities of the agents and the optimal mapping decision in each iteration. Individual agent can obtain its neighbors' position information by using its senors. The predefined formation is treated as the reference of the final formation, which the group of elliptical agents should achieve. The agents do not need to move to the fixed points in the predefined formation. They only need to find the optimal positions based on the displacements from the predefined formation. Formation controller for each elliptical agent is developed based on the its desired position, which is obtained by using its current position, its neighbors's position information and its desired position. The desired position of individual agent is calculated based on the its position in predefined formation corresponding with the optimal mapping, and its neighbors' corresponding positions in the predefined formation. The random mapping algorithm is investigated to find the optimal mapping in each iteration until the agents reach their final formation. Collision avoidance algorithm is developed based on the moving orientation and moving distance based on the agents' avoidance groups, while the self-center-based rotation algorithm is constructed to extend the minimum distances among elliptical agents.

To improve the efficiency of the collision-free formation control strategy in Chapter 2, an adaptive collision-free formation control strategy is developed in Chapter 3. The adaptive random mapping algorithm replaces the random mapping algorithm to improve efficiency and reduce the computational burden. It executes during the group moving. The adaptive mapping scheme is built based on the minimum value of the total distances corresponded to the generated mappings in each iteration, which is calculated based on the current positions and the desired positions in that iteration of all agents. The optimal mapping pool is constructed to place the optimal mappings in each iteration. After a fixed number of iterations, we can find the following optimal mappings based on the repeat rate and reacting elements relying on the optimal mapping pool. An adaptive parameter is introduced to the formation controller of individual elliptical to adapt to the variety of the number of the elements in its avoidance groups, its desired positions and the minimum distance it has. The moving length can be adjusted based on this adaptive parameter.

To accommodate a larger group of elliptical agents, Chapter 4 presents event-triggered probability-driven formation control scheme for a group of elliptical agents. The event-triggered scheme is investigated by using the collision time and the deviation time. Each elliptical agent has its own event sequence. Agents only need to receive the state and velocity information in accordance with their own event sequence. The collision time for individual agent is obtained based on the position and velocity information of its possible collision agents, while relative velocity in the direction of obstacle avoidance should be calculated based on the velocities of its current velocity and the velocity of its possible collision agents. The deviation time for each elliptical agent can be calculated based on its position and velocity information and its desired position based on the predefined formation. The probability-driven controller is developed to deal with the stuck problem which may happen during formation moving. To improve the performance of the event-triggered formation control strategy, the adaptive schemes for mapping decision and heading angle rotation are employed to find the optimal mapping, reduce moving distance of the whole group and maintain collision free in the group.

Chapter 5 attempts to implement a two-stage reconfiguration strategy for multi-agent systems. This strategy is employed to find the special formation during idle time, which can be seen as transition of the predefined formations, in order to minimize the reconfiguration time. First, the two-stage reconfiguration scheme for dot agents is proposed to be treated as a basic reconfiguration strategy. The random mapping algorithm is introduced to find the optimal mapping for each predefined formation. These optimal mappings will lead the group of agents to find their optimal special formation. To meet the practical requirements, the agents are modeled as circles or ellipses to consider overlapping problem, which may happen by using the two-stage reconfiguration strategy for dot agents. The anti-overlapping strategies are built to construct the achievable special formation based on the geometric properties of circle and ellipse.

The main contributions of this thesis are given as follows.

1 New control schemes are proposed to drive a group of elliptical agents to a predefined formation. All agents are assumed to have the same form of control law and reference formation. Only restricted communication among agents is allowed, and they can send and receive identification numbers to and from other agents in the system. The controller of each agent is established based on the midpoint derived from their neighborhood. An adaptive parameter is introduced to the formation control strategy to adapt to the variety neighboring environment of each elliptical agent. The collision among elliptical agents can be avoided by choosing optimal path and removing obstacle angles. A self-center-based rotation algorithm is also proposed to guarantee collision avoidance when two agents approach to each other.

2 The desired formation is obtained based on the displacements and predefined formation, which are obtained though a reference mapping. Agents can find their optimal mapping decisions based on the random

mapping algorithm. During each sampling interval, several possible mappings are generated and the sums of distances with corresponding agents under each possible mapping decision are calculated to be compared with the others. The shortest one will be chosen to be the optimal formation in the corresponding interval. To reduce the computational burden, an adaptive random mapping algorithm is developed based on the random mapping algorithm. It is achieved based on the repeat rate and repeating elements in each optimal mapping.

3 An event-triggered probability-driven formation control scheme is investigated for multiple elliptical agents. Each agent has its own event sequence based on the minimum collision time and the deviation time calculated by itself. Agents only need to receive the state and velocity information in accordance with their own event sequence. Probability-driven controller is established to prevent the stuck problem among agents, which may happen when two or more elliptical agents are too close to each other.

4 A two-stage reconfiguration strategy based on dot agents is proposed during idle time with a random mapping algorithm. The mapping relationship applied in this thesis is changing based on the current positions of the agents and the predefined formation. This mapping algorithm is constructed based on the minimum expected moving distance between the current positions of the group of agents and each predefined formation. To meet the practical requirements, the two-stage reconfiguration scheme is improved due to the circular shapes and elliptical shapes of the agents, which are used to find the optimal special formation without overlapping problem during idle time.

Bibliography

[1] N. Rhinehart and K. Kitani, "First-person activity forecasting from video with online inverse reinforcement learning", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–1, 2018, ISSN: 1939-3539. DOI: 10.1109/TPAMI.2018.2873794.

[2] S. Su, J. P. Hong, J. Shi, and H. S. Park, "Social behavior prediction from first person videos", *arXiv preprint arXiv:1611.09464*, 2016.

[3] S. F. Railsback and V. Grimm, *Agent-based and individual-based modeling: a practical introduction*. Princeton university press, 2019.

[4] M. P. Wellman, "Putting the agent in agent-based modeling", *Autonomous Agents and Multi-Agent Systems*, vol. 30, no. 6, pp. 1175–1189, 2016.

[5] H. Yu, P. Shi, and C.C. Lim, "Robot formation control in stealth mode with scalable team size", *International Journal of Control*, vol. 89, no. 11, pp. 2155–2168, 2016.

[6] T. Nguyen, H. M. La, T. D. Le, and M. Jafari, "Formation control and obstacle avoidance of multiple rectangular agents with limited communication ranges", *IEEE Transactions on Control of Network Systems*, vol. 4, no. 4, pp. 680–691, 2016.

[7] M. Yu, H. Wang, G. Xie, and K. Jin, "Event-triggered circle formation control for second-order-agent system", *Neurocomputing*, vol. 275, pp. 462–469, 2018.

[8] B. Jiang, M. Deghat, and B. D. O. Anderson, "Simultaneous velocity and position estimation via distance-only measurements with application to multi-agent system control", *IEEE Transactions on Automatic Control*, vol. 62, no. 2, pp. 869–875, 2017, ISSN: 0018-9286. DOI: 10.1109/TAC.2016.2558040.

[9] M. Wooldridge and N. R. Jennings, "Intelligent agents: Theory and practice", *The Knowledge Engineering Review*, vol. 10, no. 2, 115–152, 1995. DOI: 10.1017/S0269888900008122.

[10] M. H. Yamchi and R. M. Esfanjani, "Distributed predictive formation control of networked mobile robots subject to communication delay", *Robotics and Autonomous Systems*, vol. 91, pp. 194–207, 2017, ISSN: 0921-8890. DOI: https://doi.org/10.1016/j.robot.2017.01.005. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0921889016301014.

[11] A Sujil, S. K. Agarwal, and R. Kumar, "Centralized multi-agent implementation for securing critical loads in pv based microgrid", *Journal of Modern Power Systems and Clean Energy*, vol. 2, no. 1, pp. 77–86, 2014.

[12] B. Baki, M. Bouzid, A Ligęza, and A.-I. Mouaddib, "A centralized planning technique with temporal constraints and uncertainty for multi-agent systems", *Journal of Experimental and Theoretical Artificial Intelligence*, vol. 18, no. 3, pp. 331–364, 2006.

[13] T. Pereira, N. Luis, A. Moreira, D. Borrajo, M. Veloso, and S. Fernandez, "Heterogeneous multi-agent planning using actuation maps", in *2018 IEEE International Conference on Autonomous Robot Systems and Competitions (ICARSC)*, 2018, pp. 219–224. DOI: 10.1109/ICARSC.2018.8374186.

[14] G. Andreadis, P. Klazoglou, K. Niotaki, and K.-D. Bouzakis, "Classification and review of multi-agents systems in the manufacturing section", *Procedia Engineering*, vol. 69, pp. 282–290, 2014.

[15] D. Portugal and R. P. Rocha, "Cooperative multi-robot patrol with bayesian learning", *Autonomous Robots*, vol. 40, no. 5, pp. 929–953, 2016.

[16] M. T. Nguyen, L. Rodrigues, C. S. Maniu, and S. Olaru, "Discretized optimal control approach for dynamic multi-agent decentralized coverage", in *2016 IEEE International Symposium on Intelligent Control (ISIC)*, 2016, pp. 1–6. DOI: 10.1109/ISIC.2016.7579984.

[17] G. Best, O. M. Cliff, T. Patten, R. R. Mettu, and R. Fitch, "Dec-mcts: Decentralized planning for multi-robot active perception", *The International Journal of Robotics Research*, vol. 38, no. 2-3, pp. 316–337, 2019.

[18] M. Schatten, P. Grd, M. Konecki, and R. Kudelić, "Towards a formal conceptualization of organizational design techniques for large scale multi agent systems", *Procedia Technology*, vol. 15, pp. 576–585, 2014.

[19] J. Ferber and O. Gutknecht, "A meta-model for the analysis and design of organizations in multi-agent systems", in *Proceedings International Conference on Multi Agent Systems (Cat. No.98EX160)*, 1998, pp. 128–135. DOI: 10.1109/ICMAS.1998.699041.

[20] H. Du and S. Li, "Attitude synchronization for flexible spacecraft with communication delays", *IEEE Transactions on Automatic Control*, vol. 61, no. 11, pp. 3625–3630, 2016, ISSN: 2334-3303. DOI: 10.1109/TAC.2016.2525933.

[21] Z. Zhang, Y. Zheng, X. Xiao, and W. Yan, "Improved order-reduction method for cooperative tracking control of time-delayed multi-spacecraft network", *Journal of the Franklin Institute*, vol. 355, no. 5, pp. 2849–2873, 2018, ISSN: 0016-0032. DOI: https://doi.org/10.1016/j.jfranklin.2018.01.019. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0016003218300796.

[22] P. Li, "Global finite-time attitude consensus tracking control for a group of rigid spacecraft", *International Journal of Systems Science*, vol. 48, no. 13, pp. 2703–2712, 2017.

[23] J. Zhang, Q. Hu, and W. Xie, "Integral sliding mode-based attitude coordinated tracking for spacecraft formation with communication delays", *International Journal of Systems Science*, vol. 48, no. 15, pp. 3254–3266, 2017. DOI: 10.1080/00207721.2017.1371359. eprint: https://doi.org/10.1080/00207721.2017.1371359. [Online]. Available: https://doi.org/10.1080/00207721.2017.1371359.

[24] I. Shames, B. Fidan, and B. D. O. Anderson, "Close target reconnaissance with guaranteed collision avoidance", *International Journal of Robust and Nonlinear Control*, vol. 21, no. 16, pp. 1823–1840, 2011. DOI: 10.1002/rnc.1663. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/rnc.1663. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/rnc.1663.

[25] B. D. O. Anderson, B. Fidan, C. Yu, and D. Walle, "Uav formation control: Theory and application", in *Recent Advances in Learning and Control*, V. D. Blondel, S. P. Boyd, and H. Kimura, Eds., London: Springer London, 2008, pp. 15–33, ISBN: 978-1-84800-155-8.

[26] D. van der Walle, B. Fidan, A. Sutton, Changbin Yu, and B. D. O. Anderson, "Non-hierarchical uav formation control for surveillance tasks", in *2008 American Control Conference*, 2008, pp. 777–782. DOI: 10.1109/ACC.2008.4586587.

[27] K. Guo, X. Li, and L. Xie, "Ultra-wideband and odometry-based cooperative relative localization with application to multi-uav formation control", *IEEE Transactions on Cybernetics*, pp. 1–14, 2019, ISSN: 2168-2275. DOI: 10.1109/TCYB.2019.2905570.

[28] M. Dadgar, S. Jafari, and A. Hamzeh, "A PSO-based multi-robot cooperation method for target searching in unknown environments", *Neurocomputing*, vol. 177, pp. 62–74, 2016, ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2015.11.007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231215016586.

[29] Z. Ming-hui, M. Sai, and W. Xing-ru, "Design of moving system for mining searching robot", in *2017 9th International Conference on Intelligent Human-Machine Systems and Cybernetics (IHMSC)*, vol. 1, 2017, pp. 35–37. DOI: 10.1109/IHMSC.2017.15.

[30] H. M. La and W. Sheng, "Distributed sensor fusion for scalar field mapping using mobile sensor networks", *IEEE Transactions on Cybernetics*, vol. 43, no. 2, pp. 766–778, 2013, ISSN: 2168-2275. DOI: 10.1109/TSMCB.2012.2215919.

[31] H. Wang, C. Zhang, Y. Song, and B. Pang, "Master-followed multiple robots cooperation slam adapted to search and rescue environment", *International Journal of Control, Automation and Systems*, vol. 16, no. 6, pp. 2593–2608, 2018, ISSN: 2005-4092. DOI: 10.1007/s12555-017-0227-7. [Online]. Available: https://doi.org/10.1007/s12555-017-0227-7.

[32] B. Das, B. Subudhi, and B. B. Pati, "Cooperative formation control of autonomous underwater vehicles: An overview", *International Journal of Automation and computing*, vol. 13, no. 3, pp. 199–225, 2016.

[33] C. Yuan, S. Licht, and H. He, "Formation learning control of multiple autonomous underwater vehicles with heterogeneous nonlinear uncertain dynamics", *IEEE Transactions on Cybernetics*, vol. 48, no. 10, pp. 2920–2934, 2018, ISSN: 2168-2275. DOI: 10.1109/TCYB.2017.2752458.

[34] H. Li, P. Xie, and W. Yan, "Receding horizon formation tracking control of constrained underactuated autonomous underwater vehicles", *IEEE Transactions on Industrial Electronics*, vol. 64, no. 6, pp. 5004–5013, 2017, ISSN: 1557-9948. DOI: 10.1109/TIE.2016.2589921.

[35] J. Wang, C. Wang, Y. Wei, and C. Zhang, "Neuroadaptive sliding mode formation control of autonomous underwater vehicles with uncertain dynamics", *IEEE Systems Journal*, pp. 1–9, 2019, ISSN: 2373-7816. DOI: 10.1109/JSYST.2019.2938315.

[36] W. Ren and Y. Cao, *Distributed coordination of multi-agent networks: emergent problems, models, and issues*. Springer Science & Business Media, 2010.

[37] V. E. Pryanichnikov, A. A. Aryskin, S. R. Eprikov, K. B. Kirsanov, R. V. Khelemendik, A. Y. Ksenzenko, E. A. Prysev, and A. S. Travushkin, "Technology of multi-agent control for industrial automation with logical processing of contradictions", *Annals of DAAAM & Proceedings*, pp. 1202–1208, 2018.

[38] V. N. Coelho, M. W. Cohen, I. M. Coelho, N. Liu, and F. G. Guimarães, "Multi-agent systems applied for energy systems integration: State-of-the-art applications and trends in microgrids", *Applied Energy*, vol. 187, pp. 820 –832, 2017, ISSN: 0306-2619. DOI: https://doi.org/10.1016/j.apenergy.2016.10.056. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0306261916315008.

[39] O. J. Shukla, A. Joshi, G. Soni, and R. Kumar, "Analysis of critical drivers affecting implementation of agent technology in a manufacturing system", *Journal of Industrial Engineering International*, vol. 15, no. 2, pp. 303–313, 2019, ISSN: 2251-712X. DOI: 10.1007/s40092-018-0293-3. [Online]. Available: https://doi.org/10.1007/s40092-018-0293-3.

[40] K. Jose and D. K. Pratihar, "Task allocation and collision-free path planning of centralized multi-robots system for industrial plant inspection using heuristic methods", *Robotics and Autonomous Systems*, vol. 80, pp. 34–42, 2016, ISSN: 0921-8890. DOI: https://doi.org/10.1016/j.robot.2016.02.003. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0921889016000282.

[41] C. Wu, R. Zeng, J. Pan, C. C. L. Wang, and Y. Liu, “Plant phenotyping by deep-learning-based planner for multi-robots”, *IEEE Robotics and Automation Letters*, vol. 4, no. 4, pp. 3113–3120, 2019, ISSN: 2377-3774. DOI: 10.1109/LRA.2019.2924125.

[42] Z. Miao, J. Yu, J. Ji, and J. Zhou, “Multi-objective region reaching control for a swarm of robots”, *Automatica*, vol. 103, pp. 81–87, 2019, ISSN: 0005-1098. DOI: https://doi.org/10.1016/j.automatica.2019.01.017. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S000510981930024X.

[43] H. Wang, D. Guo, X. Liang, W. Chen, G. Hu, and K. K. Leang, “Adaptive vision-based leader–follower formation control of mobile robots”, *IEEE Transactions on Industrial Electronics*, vol. 64, no. 4, pp. 2893–2902, 2017, ISSN: 1557-9948. DOI: 10.1109/TIE.2016.2631514.

[44] H. Xiao, Z. Li, and C. L. Philip Chen, “Formation control of leader–follower mobile robots’ systems using model predictive control based on neural-dynamic optimization”, *IEEE Transactions on Industrial Electronics*, vol. 63, no. 9, pp. 5752–5762, 2016, ISSN: 1557-9948. DOI: 10.1109/TIE.2016.2542788.

[45] X. Liang, H. Wang, Y. Liu, W. Chen, and T. Liu, “Formation control of nonholonomic mobile robots without position and velocity measurements”, *IEEE Transactions on Robotics*, vol. 34, no. 2, pp. 434–446, 2018, ISSN: 1941-0468. DOI: 10.1109/TRO.2017.2776304.

[46] Qin Chen and J. Y. S. Luh, “Coordination and control of a group of small mobile robots”, in *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, 1994, 2315–2320 vol.3. DOI: 10.1109/ROBOT.1994.350940.

[47] K. L. Soon, J. M.-Y. Lim, and R. Parthiban, “Coordinated traffic light control in cooperative green vehicle routing for pheromone-based multi-agent systems”, *Applied Soft Computing*, vol. 81, p. 105 486, 2019, ISSN: 1568-4946. DOI: https://doi.org/10.1016/j.asoc.2019.105486. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S156849461930256X.

[48] T. Chu, J. Wang, L. Codecà, and Z. Li, “Multi-agent deep reinforcement learning for large-scale traffic signal control”, *IEEE Transactions on Intelligent Transportation Systems*, pp. 1–10, 2019, ISSN: 1558-0016. DOI: 10.1109/TITS.2019.2901791.

[49] S. Aminzadegan, M. Tamannaei, and M. Rasti-Barzoki, “Multi-agent supply chain scheduling problem by considering resource allocation and transportation”, *Computers & Industrial Engineering*, vol. 137, p. 106 003, 2019, ISSN: 0360-8352. DOI: https://doi.org/10.1016/j.cie.2019.106003. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0360835219304619.

[50] M. Dotoli, H. Zgaya, C. Russo, and S. Hammadi, "A multi-agent advanced traveler information system for optimal trip planning in a co-modal framework", *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 9, pp. 2397–2412, 2017, ISSN: 1558-0016. DOI: 10.1109/TITS.2016.2645278.

[51] L. B. de Oliveira and E. Camponogara, "Multi-agent model predictive control of signaling split in urban traffic networks", *Transportation Research Part C: Emerging Technologies*, vol. 18, no. 1, pp. 120–139, 2010, Information/Communication Technologies and Travel Behaviour Agents in Traffic and Transportation, ISSN: 0968-090X. DOI: https://doi.org/10.1016/j.trc.2009.04.022. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0968090X09000540.

[52] M. M. Ibrahim and W. A. Omran, "A decentralized coordination strategy for voltage regulation of active distribution networks", in *2016 IEEE PES Innovative Smart Grid Technologies Conference Europe (ISGT-Europe)*, 2016, pp. 1–6. DOI: 10.1109/ISGTEurope.2016.7856227.

[53] N. Jennings, L. Varga, R. Aarnts, J. Fuchs, and P. Skarek, "Transforming standalone expert systems into a community of cooperating agents", *Engineering Applications of Artificial Intelligence*, vol. 6, no. 4, pp. 317–331, 1993, ISSN: 0952-1976. DOI: https://doi.org/10.1016/0952-1976(93)90016-Q. [Online]. Available: http://www.sciencedirect.com/science/article/pii/095219769390016Q.

[54] X. Wang, S. Li, X. Yu, and J. Yang, "Distributed active anti-disturbance consensus for leader-follower higher-order multi-agent systems with mismatched disturbances", *IEEE Transactions on Automatic Control*, vol. 62, no. 11, pp. 5795–5801, 2017, ISSN: 2334-3303. DOI: 10.1109/TAC.2016.2638966.

[55] B. Tian, Z. Zuo, and H. Wang, "Leader–follower fixed-time consensus of multi-agent systems with high-order integrator dynamics", *International Journal of Control*, vol. 90, no. 7, pp. 1420–1427, 2017. DOI: 10.1080/00207179.2016.1207101. eprint: https://doi.org/10.1080/00207179.2016.1207101. [Online]. Available: https://doi.org/10.1080/00207179.2016.1207101.

[56] P. Shi and Q. K. Shen, "Observer-based leader-following consensus of uncertain nonlinear multi-agent systems", *International Journal of Robust and Nonlinear Control*, vol. 27, no. 17, pp. 3794–3811, 2017. DOI: 10.1002/rnc.3766. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/rnc.3766. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/rnc.3766.

[57] J. Lü, F. Chen, and G. Chen, "Nonsmooth leader-following formation control of nonidentical multi-agent systems with directed communication topologies", *Automatica*, vol. 64, pp. 112 –120, 2016, ISSN: 0005-1098. DOI: https://doi.org/10.1016/j.automatica.2015.11.004. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0005109815004689.

[58] P. Gong, "Distributed tracking of heterogeneous nonlinear fractional-order multi-agent systems with an unknown leader", *Journal of the Franklin Institute*, vol. 354, no. 5, pp. 2226–2244, 2017, ISSN: 0016-0032. DOI: https://doi.org/10.1016/j.jfranklin.2017.01.001. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0016003217300042.

[59] J. Wang and M. Xin, "Integrated optimal formation control of multiple unmanned aerial vehicles", *IEEE Transactions on Control Systems Technology*, vol. 21, no. 5, pp. 1731–1744, 2013, ISSN: 2374-0159. DOI: 10.1109/TCST.2012.2218815.

[60] S. Kim and Y. Kim, "Optimum design of three-dimensional behavioural decentralized controller for uav formation flight", *Engineering Optimization*, vol. 41, no. 3, pp. 199–224, 2009. DOI: 10.1080/03052150802406532. eprint: https://doi.org/10.1080/03052150802406532. [Online]. Available: https://doi.org/10.1080/03052150802406532.

[61] S.-k. Pang, Y.-h. Li, and H. Yi, "Joint formation control with obstacle avoidance of towfish and multiple autonomous underwater vehicles based on graph theory and the null-space-based method", *Sensors*, vol. 19, no. 11, 2019, ISSN: 1424-8220. DOI: 10.3390/s19112591. [Online]. Available: https://www.mdpi.com/1424-8220/19/11/2591.

[62] G. Lee and D. Chwa, "Decentralized behavior-based formation control of multiple robots considering obstacle avoidance", *Intelligent Service Robotics*, vol. 11, no. 1, pp. 127–138, 2018, ISSN: 1861-2784. DOI: 10.1007/s11370-017-0240-y. [Online]. Available: https://doi.org/10.1007/s11370-017-0240-y.

[63] J. Huang, N. Zhou, and M. Cao, "Adaptive fuzzy behavioral control of second-order autonomous agents with prioritized missions: Theory and experiments", *IEEE Transactions on Industrial Electronics*, vol. 66, no. 12, pp. 9612–9622, 2019, ISSN: 1557-9948. DOI: 10.1109/TIE.2019.2892669.

[64] Z. Sun and Y. Xia, "Receding horizon tracking control of unicycle-type robots based on virtual structure", *International Journal of Robust and Nonlinear Control*, vol. 26, no. 17, pp. 3900–3918, 2016. DOI: 10.1002/rnc.3555. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/rnc.3555. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/rnc.3555.

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

[66] D. Zhou, Z. Wang, and M. Schwager, "Agile coordination and assistive collision avoidance for quadrotor swarms using virtual structures", *IEEE Transactions on Robotics*, vol. 34, no. 4, pp. 916–923, 2018, ISSN: 1941-0468. DOI: 10.1109/TRO.2018.2857477.

[67] L. Dong, Y. Chen, and X. Qu, "Formation control strategy for nonholonomic intelligent vehicles based on virtual structure and consensus approach", *Procedia Engineering*, vol. 137, pp. 415–424, 2016, Green Intelligent Transportation System and Safety, ISSN: 1877-7058. DOI: https://doi.org/10.1016/j.proeng.2016.01.276. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S1877705816003039.

[68] B.-L. Cong, X.-D. Liu, and Z. Chen, "Distributed attitude synchronization of formation flying via consensus-based virtual structure", *Acta Astronautica*, vol. 68, no. 11, pp. 1973–1986, 2011, ISSN: 0094-5765. DOI: https://doi.org/10.1016/j.actaastro.2010.11.014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0094576510004303.

[69] G. Guo and W. Yue, "Autonomous platoon control allowing range-limited sensors", *IEEE Transactions on Vehicular Technology*, vol. 61, no. 7, pp. 2901–2912, 2012, ISSN: 1939-9359. DOI: 10.1109/TVT.2012.2203362.

[70] H. Yu, P. Shi, and C.C. Lim, "Robot formation control in stealth mode with scalable team size", *International Journal of Control*, vol. 89, no. 11, pp. 2155–2168, 2016. DOI: 10.1080/00207179.2016.1149887. eprint: https://doi.org/10.1080/00207179.2016.1149887. [Online]. Available: https://doi.org/10.1080/00207179.2016.1149887.

[71] K. D. Do, "Bounded controllers for formation stabilization of mobile agents with limited sensing ranges", *IEEE Transactions on Automatic Control*, vol. 52, no. 3, pp. 569–576, 2007, ISSN: 2334-3303. DOI: 10.1109/TAC.2007.892382.

[72] M. Okamoto and M. R. Akella, "Avoiding the local-minimum problem in multi-agent systems with limited sensing and communication", *International Journal of Systems Science*, vol. 47, no. 8, pp. 1943–1952, 2016. DOI: 10.1080/00207721.2014.965239. eprint: https://doi.org/10.1080/00207721.2014.965239. [Online]. Available: https://doi.org/10.1080/00207721.2014.965239.

[73] X. Liu, S. S. Ge, C.-H. Goh, and Y. Li, "Event-triggered coordination for formation tracking control in constrained space with limited communication", *IEEE Transactions on Cybernetics*, vol. 49, no. 3, pp. 1000–1011, 2019, ISSN: 2168-2275. DOI: 10.1109/TCYB.2018.2794139.

[74] T. Nguyen, H. M. La, T. D. Le, and M. Jafari, "Formation control and obstacle avoidance of multiple rectangular agents with limited communication ranges", *IEEE Transactions on Control of Network Systems*, vol. 4, no. 4, pp. 680–691, 2017, ISSN: 2372-2533. DOI: 10.1109/TCNS.2016.2542978.

[75] P. Shi and Q. Shen, "Cooperative control of multi-agent systems with unknown state-dependent controlling effects", *IEEE Transactions on Automation Science and Engineering*, vol. 12, no. 3, pp. 827–834, 2015, ISSN: 1558-3783. DOI: 10.1109/TASE.2015.2403261.

[76] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays", *IEEE Transactions on Automatic Control*, vol. 49, no. 9, pp. 1520–1533, 2004, ISSN: 2334-3303. DOI: 10.1109/TAC.2004.834113.

[77] P. Lin and Y. Jia, "Consensus of second-order discrete-time multi-agent systems with nonuniform time-delays and dynamically changing topologies", *Automatica*, vol. 45, no. 9, pp. 2154–2158, 2009, ISSN: 0005-1098. DOI: https://doi.org/10.1016/j.automatica.2009.05.002. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0005109809002301.

[78] Y. Liu, H. Yu, C.C. Lim, and P. Shi, "Steady formation analysis on multi-robot systems", *Journal of Control and Decision*, vol. 4, no. 1, pp. 12–31, 2017. DOI: 10.1080/23307706.2016.1254073. eprint: https://doi.org/10.1080/23307706.2016.1254073. [Online]. Available: https://doi.org/10.1080/23307706.2016.1254073.

[79] X. Dong, Y. Zhou, Z. Ren, and Y. Zhong, "Time-varying formation control for unmanned aerial vehicles with switching interaction topologies", *Control Engineering Practice*, vol. 46, pp. 26–36, 2016, ISSN: 0967-0661. DOI: https://doi.org/10.1016/j.conengprac.2015.10.001. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0967066115300289.

[80] ——, "Time-varying formation tracking for second-order multi-agent systems subjected to switching topologies with application to quadrotor formation flying", *IEEE Transactions on Industrial Electronics*, vol. 64, no. 6, pp. 5014–5024, 2017, ISSN: 1557-9948. DOI: 10.1109/TIE.2016.2593656.

[81] S. He, M. Wang, S.-L. Dai, and F. Luo, "Leader–follower formation control of usvs with prescribed performance and collision avoidance", *IEEE Transactions on Industrial Informatics*, vol. 15, no. 1, pp. 572–581, 2019, ISSN: 1941-0050. DOI: 10.1109/TII.2018.2839739.

[82] L. Dai, Q. Cao, Y. Xia, and Y. Gao, "Distributed mpc for formation of multi-agent systems with collision avoidance and obstacle avoidance", *Journal of the Franklin Institute*, vol. 354, no. 4, pp. 2068–2085, 2017, ISSN: 0016-0032. DOI: https://doi.org/10.1016/j.jfranklin.2016.12.021. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0016003216304926.

[83] Z. Peng, D. Wang, T. Li, and M. Han, "Output-feedback cooperative formation maneuvering of autonomous surface vehicles with connectivity preservation and collision avoidance", *IEEE Transactions on Cybernetics*, pp. 1–9, 2019, ISSN: 2168-2275. DOI: 10.1109/TCYB.2019.2914717.

[84] A. Mondal, C. Bhowmick, L. Behera, and M. Jamshidi, "Trajectory tracking by multiple agents in formation with collision avoidance and connectivity assurance", *IEEE Systems Journal*, vol. 12, no. 3, pp. 2449–2460, 2018, ISSN: 2373-7816. DOI: 10.1109/JSYST.2017.2778063.

[85] L. Chen, Y. Guo, C. Li, and J. Huang, "Satellite formation-containment flying control with collision avoidance", *Journal of Aerospace Information Systems*, vol. 15, no. 5, pp. 253–270, 2018. DOI: 10.2514/1.I010588. eprint: https://doi.org/10.2514/1.I010588. [Online]. Available: https://doi.org/10.2514/1.I010588.

[86] M. Hosseinzadeh Yamchi and R. Mahboobi Esfanjani, "Formation control of networked mobile robots with guaranteed obstacle and collision avoidance", *Robotica*, vol. 35, no. 6, 1365–1377, 2017. DOI: 10.1017/S0263574716000102.

[87] W. Gao, Z.-P. Jiang, F. L. Lewis, and Y. Wang, "Leader-to-formation stability of multiagent systems: An adaptive optimal control approach", *IEEE Transactions on Automatic Control*, vol. 63, no. 10, pp. 3581–3587, 2018, ISSN: 2334-3303. DOI: 10.1109/TAC.2018.2799526.

[88] J. Yu, X. Dong, Q. Li, J. Lü, and Z. Ren, "Fully adaptive practical time-varying output formation tracking for high-order nonlinear stochastic multiagent system with multiple leaders", *IEEE Transactions on Cybernetics*, pp. 1–13, 2019, ISSN: 2168-2275. DOI: 10.1109/TCYB.2019.2956316.

[89] X. Li and D. Zhu, "An adaptive SOM neural network method for distributed formation control of a group of AUVs", *IEEE Transactions on Industrial Electronics*, vol. 65, no. 10, pp. 8260–8270, 2018, ISSN: 1557-9948. DOI: 10.1109/TIE.2018.2807368.

[90] Y. Wang, Y. Song, and W. Ren, "Distributed adaptive finite-time approach for formation–containment control of networked nonlinear systems under directed topology", *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 7, pp. 3164–3175, 2018, ISSN: 2162-2388. DOI: 10.1109/TNNLS.2017.2714187.

[91] L. He, X. Sun, and Y. Lin, "Distributed adaptive control for time-varying formation tracking of a class of networked nonlinear systems", *International Journal of Control*, vol. 90, no. 7, pp. 1319–1326, 2017. DOI: 10.1080/00207179.2016.1205757. eprint: https://doi.org/10.1080/00207179.2016.1205757. [Online]. Available: https://doi.org/10.1080/00207179.2016.1205757.

[92] Z. Peng, S. Yang, G. Wen, A. Rahmani, and Y. Yu, "Adaptive distributed formation control for multiple nonholonomic wheeled mobile robots", *Neurocomputing*, vol. 173, pp. 1485–1494, 2016, ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2015.09.022. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231215013302.

[93] P. Flocchini, G. Prencipe, N. Santoro, and G. Viglietta, "Distributed computing by mobile robots: Uniform circle formation", *Distributed Computing*, vol. 30, no. 6, pp. 413–457, 2017, ISSN: 1432-0452. DOI: 10.1007/s00446-016-0291-x. [Online]. Available: https://doi.org/10.1007/s00446-016-0291-x.

[94] K. Do, "Formation control of multiple elliptical agents with limited sensing ranges", *Automatica*, vol. 48, no. 7, pp. 1330–1338, 2012, ISSN: 0005-1098. DOI: https://doi.org/10.1016/j.automatica.2012.04.005. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S000510981200163X.

[95] ——, "Practical formation control of multiple underactuated ships with limited sensing ranges", *Robotics and Autonomous Systems*, vol. 59, no. 6, pp. 457–471, 2011, ISSN: 0921-8890. DOI: https://doi.org/10.1016/j.robot.2011.03.003. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0921889011000376.

[96] A. Best, S. Narang, and D. Manocha, "Real-time reciprocal collision avoidance with elliptical agents", in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 298–305. DOI: 10.1109/ICRA.2016.7487148.

[97] K. J. Åström and B. Bernhardsson, "Comparison of periodic and event based sampling for first-order stochastic systems", in *Proceedings of the 14th IFAC World congress*, Citeseer, vol. 11, 1999, pp. 301–306.

[98] D. V. Dimarogonas, E. Frazzoli, and K. H. Johansson, "Distributed event-triggered control for multi-agent systems", *IEEE Transactions on Automatic Control*, vol. 57, no. 5, pp. 1291–1297, 2012, ISSN: 2334-3303. DOI: 10.1109/TAC.2011.2174666.

[99] A. Sinha and R. K. Mishra, "Consensus in first order nonlinear heterogeneous multi-agent systems with event-based sliding mode control", *International Journal of Control*, vol. 0, no. 0, pp. 1–14, 2018. DOI: 10.1080/00207179.2018.1531147. eprint: https://doi.org/10.1080/00207179.2018.1531147. [Online]. Available: https://doi.org/10.1080/00207179.2018.1531147.

[100] Q. Lü, H. Li, and D. Xia, "Distributed optimization of first-order discrete-time multi-agent systems with event-triggered communication", *Neurocomputing*, vol. 235, pp. 255–263, 2017, ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2017.01.021. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231217300528.

[101] X. Shi, R. Zheng, Z. Lin, and G. Yan, "Consensus of first-order multi-agent systems under event-triggered communication", in *2018 Chinese Control And Decision Conference (CCDC)*, 2018, pp. 4679–4683. DOI: 10.1109/CCDC.2018.8407940.

[102] Q. Zhou, D. Wang, and W. Zhu, "Consensus of first-order multi-agent systems via adaptive event-based impulsive control", in *2018 37th Chinese Control Conference (CCC)*, 2018, pp. 6996–7000. DOI: 10.23919/ChiCC.2018.8483886.

[103] M.-M. Duan, C.-L. Liu, and F. Liu, "Event-triggered consensus seeking of heterogeneous first-order agents with input delay", *IEEE Access*, vol. 5, pp. 5215–5223, 2017, ISSN: 2169-3536. DOI: 10.1109/ACCESS.2017.2696026.

[104] W. Zhu, H. Pu, D. Wang, and H. Li, "Event-based consensus of second-order multi-agent systems with discrete time", *Automatica*, vol. 79, pp. 78–83, 2017, ISSN: 0005-1098. DOI: https://doi.org/10.1016/j.automatica.2017.01.042. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0005109817300523.

[105] H. Xia, W. X. Zheng, and J. Shao, "Event-triggered containment control for second-order multi-agent systems with sampled position data", *ISA Transactions*, vol. 73, pp. 91–99, 2018, ISSN: 0019-0578. DOI: https://doi.org/10.1016/j.isatra.2017.11.001. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0019057817306092.

[106] N. Mu, X. Liao, and T. Huang, "Consensus of second-order multi-agent systems with random sampling via event-triggered control", *Journal of the Franklin Institute*, vol. 353, no. 6, pp. 1423–1435, 2016, ISSN: 0016-0032. DOI: https://doi.org/10.1016/j.jfranklin.2016.01.014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0016003216000405.

[107] J. Dai and G. Guo, "Event-based consensus for second-order multi-agent systems with actuator saturation under fixed and markovian switching topologies", *Journal of the Franklin Institute*, vol. 354, no. 14, pp. 6098–6118, 2017, ISSN: 0016-0032. DOI: https://doi.org/10.1016/j.jfranklin.2017.07.011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0016003217303253.

[108] M. Cao, F. Xiao, and L. Wang, "Event-based second-order consensus control for multi-agent systems via synchronous periodic event detection", *IEEE Transactions on Automatic Control*, vol. 60, no. 9, pp. 2452–2457, 2015, ISSN: 2334-3303. DOI: 10.1109/TAC.2015.2390553.

[109] W. Hu, L. Liu, and G. Feng, "Consensus of linear multi-agent systems by distributed event-triggered strategy", *IEEE Transactions on Cybernetics*, vol. 46, no. 1, pp. 148–157, 2016, ISSN: 2168-2275. DOI: 10.1109/TCYB.2015.2398892.

[110] W. Hu, L. Liu, and G. Feng, "Output consensus of heterogeneous linear multi-agent systems by distributed event-triggered/self-triggered strategy", *IEEE Transactions on Cybernetics*, vol. 47, no. 8, pp. 1914–1924, 2017, ISSN: 2168-2267. DOI: 10.1109/TCYB.2016.2602327.

[111] E. Garcia, Y. Cao, and D. W. Casbeer, "Periodic event-triggered synchronization of linear multi-agent systems with communication delays", *IEEE Transactions on Automatic Control*, vol. 62, no. 1, pp. 366–371, 2017, ISSN: 2334-3303. DOI: 10.1109/TAC.2016.2555484.

[112] E. Garcia, Y. Cao, and D. W. Casbeer, "Decentralized event-triggered consensus with general linear dynamics", *Automatica*, vol. 50, no. 10, pp. 2633–2640, 2014, ISSN: 0005-1098. DOI: https://doi.org/10.1016/j.automatica.2014.08.024. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0005109814003409.

[113] W. Zou and Z. Xiang, "Event-triggered distributed containment control of heterogeneous linear multi-agent systems by an output regulation approach", *International Journal of Systems Science*, vol. 48, no. 10, pp. 2041–2054, 2017. DOI: 10.1080/00207721.2017.1309595. eprint: https://doi.org/10.1080/00207721.2017.1309595. [Online]. Available: https://doi.org/10.1080/00207721.2017.1309595.

[114] P. Yu, L. Ding, Z.-W. Liu, and Z.-H. Guan, "A distributed event-triggered transmission strategy for exponential consensus of general linear multi-agent systems with directed topology", *Journal of the Franklin Institute*, vol. 352, no. 12, pp. 5866–5881, 2015, ISSN: 0016-0032. DOI: https://doi.org/10.1016/j.jfranklin.2015.10.014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0016003215003932.

[115] D. Xie, S. Xu, Y. Chu, and Y. Zou, "Event-triggered average consensus for multi-agent systems with nonlinear dynamics and switching topology", *Journal of the Franklin Institute*, vol. 352, no. 3, pp. 1080–1098, 2015, ISSN: 0016-0032. DOI: https://doi.org/10.1016/j.jfranklin.2014.11.004. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0016003214003202.

[116] N. Huang, Z. Duan, G. Wen, and Y. Zhao, "Event-triggered consensus tracking of multi-agent systems with lur'e nonlinear dynamics", *International Journal of Control*, vol. 89, no. 5, pp. 1025–1037, 2016. DOI: 10.1080/00207179.2015.1114146. eprint: https://doi.org/10.1080/00207179.2015.1114146. [Online]. Available: https://doi.org/10.1080/00207179.2015.1114146.

[117] H. Li, G. Chen, T. Huang, W. Zhu, and L. Xiao, "Event-triggered consensus in nonlinear multi-agent systems with nonlinear dynamics and directed network topology", *Neurocomputing*, vol. 185, pp. 105–112, 2016, ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2015.12.047. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231215019840.

[118] W. Zou and Z. Xiang, "Event-triggered containment control of second-order nonlinear multi-agent systems", *Journal of the Franklin Institute*, vol. 356, no. 17, pp. 10 421–10 438, 2019, Special Issue on Distributed Event-Triggered Control, Estimation, and Optimization, ISSN: 0016-0032. DOI: https://doi.org/10.1016/j.jfranklin.2018.05.060. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0016003218303922.

[119] W. Hu, C. Yang, T. Huang, and W. Gui, "A distributed dynamic event-triggered control approach to consensus of linear multiagent systems with directed networks", *IEEE Transactions on Cybernetics*, vol. 50, no. 2, pp. 869–874, 2020, ISSN: 2168-2275. DOI: 10.1109/TCYB.2018.2868778.

[120] X. Ge and Q. Han, "Distributed formation control of networked multi-agent systems using a dynamic event-triggered communication mechanism", *IEEE Transactions on Industrial Electronics*, vol. 64, no. 10, pp. 8118–8127, 2017, ISSN: 1557-9948. DOI: 10.1109/TIE.2017.2701778.

[121] Z. Zhang and L. Wang, "Distributed integral-type event-triggered synchronization of multiagent systems", *International Journal of Robust and Nonlinear Control*, vol. 28, no. 14, pp. 4175–4187, 2018. DOI: 10.1002/rnc.4227. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/rnc.4227. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/rnc.4227.

[122] Z. Zhang, J. Lunze, and L. Wang, "Integral-based event-triggered control for multi-agent systems with general linear dynamics", *International Journal of Control*, vol. 0, no. 0, pp. 1–10, 2018. DOI: 10.1080/00207179.2018.1486042. eprint: https://doi.org/10.1080/00207179.2018.1486042. [Online]. Available: https://doi.org/10.1080/00207179.2018.1486042.

[123] A. Wang, "Event-based consensus control for single-integrator networks with communication time delays", *Neurocomputing*, vol. 173, pp. 1715–1719, 2016, ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2015.09.044. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0925231215013521.

[124] X. Liu, C. Du, P. Lu, and D. Yang, "Distributed event-triggered feedback consensus control with state-dependent threshold for general linear multi-agent systems", *International Journal of Robust and Nonlinear Control*, vol. 27, no. 15, pp. 2589–2609, 2017. DOI: 10.1002/rnc.3700. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1002/rnc.3700. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/rnc.3700.

[125] B. Qi, B.-T. Cui, and X.-Y. Lou, "State-dependent event-triggered control of multi-agent systems", *Chinese Physics B*, vol. 23, no. 11, p. 110 501, 2014. DOI: 10.1088/1674-1056/23/11/110501. [Online]. Available: https://doi.org/10.1088%2F1674-1056%2F23%2F11%2F110501.

[126] A. Anta and P. Tabuada, "To sample or not to sample: Self-triggered control for nonlinear systems", *IEEE Transactions on Automatic Control*, vol. 55, no. 9, pp. 2030–2042, 2010, ISSN: 2334-3303. DOI: 10.1109/TAC.2010.2042980.

[127] G. S. Seyboth, D. V. Dimarogonas, and K. H. Johansson, "Event-based broadcasting for multi-agent average consensus", *Automatica*, vol. 49, no. 1, pp. 245–252, 2013, ISSN: 0005-1098. DOI: https://doi.org/10.1016/j.automatica.2012.08.042. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0005109812004852.

[128] W. Zhang, Y. Tang, Y. Liu, and J. Kurths, "Event-triggering containment control for a class of multi-agent networks with fixed and switching topologies", *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 64, no. 3, pp. 619–629, 2017, ISSN: 1558-0806. DOI: 10.1109/TCSI.2016.2618944.

[129] G. Shi and K. H. Johansson, "Multi-agent robust consensus-part ii: Application to distributed event-triggered coordination", in *2011 50th IEEE Conference on Decision and Control and European Control Conference*, 2011, pp. 5738–5743. DOI: 10.1109/CDC.2011.6160958.

[130] A. Wang, B. Mu, and Y. Shi, "Consensus control for a multi-agent system with integral-type event-triggering condition and asynchronous periodic detection", *IEEE Transactions on Industrial Electronics*, vol. 64, no. 7, pp. 5629–5639, 2017, ISSN: 1557-9948. DOI: 10.1109/TIE.2017.2677312.

[131] H. Yu and P. J. Antsaklis, "Output synchronization of networked passive systems with event-driven communication", *IEEE Transactions on Automatic Control*, vol. 59, no. 3, pp. 750–756, 2014, ISSN: 2334-3303. DOI: 10.1109/TAC.2013.2274704.

[132] W. Zhu and Z. Jiang, "Event-based leader-following consensus of multi-agent systems with input time delay", *IEEE Transactions on Automatic Control*, vol. 60, no. 5, pp. 1362–1367, 2015, ISSN: 2334-3303. DOI: 10.1109/TAC.2014.2357131.

[133] B. Wei, F. Xiao, and M. Dai, "Edge event-triggered synchronization in networks of coupled harmonic oscillators", *IEEE Transactions on Cybernetics*, vol. 47, no. 12, pp. 4162–4168, 2017, ISSN: 2168-2275. DOI: 10.1109/TCYB.2016.2601628.

[134] X. Yin, D. Yue, and S. Hu, "Distributed event-triggered control of discrete-time heterogeneous multi-agent systems", *Journal of the Franklin Institute*, vol. 350, no. 3, pp. 651–669, 2013, ISSN: 0016-0032. DOI: https://doi.org/10.1016/j.jfranklin.2012.12.015. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0016003212003201.

[135] X. Meng, L. Xie, and Y. C. Soh, "Asynchronous periodic event-triggered consensus for multi-agent systems", *Automatica*, vol. 84, pp. 214–220, 2017, ISSN: 0005-1098. DOI: https://doi.org/10.1016/j.automatica.2017.07.008. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0005109817303412.

[136] D. Liuzza, D. V. Dimarogonas, M. di Bernardo, and K. H. Johansson, "Distributed model based event-triggered control for synchronization of multi-agent systems", *Automatica*, vol. 73, pp. 1–7, 2016, ISSN: 0005-1098. DOI: https://doi.org/10.1016/j.automatica.2016.06.011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0005109816302370.

[137] H. Zhang, G. Feng, H. Yan, and Q. Chen, "Observer-based output feedback event-triggered control for consensus of multi-agent systems", *IEEE Transactions on Industrial Electronics*, vol. 61, no. 9, pp. 4885–4894, 2014, ISSN: 1557-9948. DOI: 10.1109/TIE.2013.2290757.

[138] J. Almeida, C. Silvestre, and A. Pascoal, "Synchronization of multi-agent systems using event-triggered and self-triggered broadcasts", *IEEE Transactions on Automatic Control*, vol. 62, no. 9, pp. 4741–4746, 2017, ISSN: 2334-3303. DOI: 10.1109/TAC.2017.2671029.

[139] F. Liao, R. Teo, J. L. Wang, X. Dong, F. Lin, and K. Peng, "Distributed formation and reconfiguration control of VTOL UAVs", *IEEE Transactions on Control Systems Technology*, vol. 25, no. 1, pp. 270–277, 2017, ISSN: 2374-0159. DOI: 10.1109/TCST.2016.2547952.

[140] P. Dasgupta and K. Cheng, "Dynamic multi-robot team reconfiguration using weighted voting games", *Journal of Experimental & Theoretical Artificial Intelligence*, vol. 28, no. 4, pp. 607–628, 2016.

[141] J. L. Nandagopal, A. Vivek, and A. Purushothaman, "Compound optimal docking of multiple spacecrafts with formation reconfiguration", in *2018 4th International Conference on Control, Automation and Robotics (ICCAR)*, 2018, pp. 252–256. DOI: 10.1109/ICCAR.2018.8384679.

[142] M. Li, H. Peng, and W. Zhong, "Optimal control of loose spacecraft formations near libration points with collision avoidance", *Nonlinear Dynamics*, vol. 83, no. 4, pp. 2241–2261, 2016.

[143] L. A. Sobiesiak and C. J. Damaren, "Lorentz-augmented spacecraft formation reconfiguration", *IEEE Transactions on Control Systems Technology*, vol. 24, no. 2, pp. 514–524, 2016, ISSN: 2374-0159. DOI: 10.1109/TCST.2015.2461593.

[144] A. Ajorlou, K. Moezzi, A. G. Aghdam, S. Tafazoli, and S. G. Nersesov, "Two-stage energy-optimal formation reconfiguration strategy", *Automatica*, vol. 48, no. 10, pp. 2587–2591, 2012, ISSN: 0005-1098. DOI: https://doi.org/10.1016/j.automatica.2012.06.059. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0005109812003251.

[145] A. Ajorlou, K. Moezzi, A. G. Aghdam, and S. G. Nersesov, "Two-stage time-optimal formation reconfiguration strategy", *Systems & Control Letters*, vol. 62, no. 6, pp. 496–502, 2013.

[146] J. M. Solanki and N. N. Schulz, "Using intelligent multi-agent systems for shipboard power systems reconfiguration", in *Proceedings of the 13th International Conference on, Intelligent Systems Application to Power Systems*, 2005, 3 pp.–. DOI: 10.1109/ISAP.2005.1599265.

[147] A. Arif, Z. Wang, J. Wang, and C. Chen, "Power distribution system outage management with co-optimization of repairs, reconfiguration, and dg dispatch", *IEEE Transactions on Smart Grid*, vol. 9, no. 5, pp. 4109–4118, 2018, ISSN: 1949-3061. DOI: 10.1109/TSG.2017.2650917.

[148] A. M. Imran, M. Kowsalya, and D. Kothari, "A novel integration technique for optimal network reconfiguration and distributed generation placement in power distribution networks", *International Journal of Electrical Power & Energy Systems*, vol. 63, pp. 461–472, 2014, ISSN: 0142-0615. DOI: https://doi.org/10.1016/j.ijepes.2014.06.011. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0142061514003585.

[149] B. Venkatesh, R. Ranjan, and H. B. Gooi, "Optimal reconfiguration of radial distribution systems to maximize loadability", *IEEE Transactions on Power Systems*, vol. 19, no. 1, pp. 260–266, 2004, ISSN: 1558-0679. DOI: 10.1109/TPWRS.2003.818739.

[150] A. M. Eldurssi and R. M. O'Connell, "A fast nondominated sorting guided genetic algorithm for multi-objective power distribution system reconfiguration problem", *IEEE Transactions on Power Systems*, vol. 30, no. 2, pp. 593–601, 2015, ISSN: 1558-0679. DOI: 10.1109/TPWRS.2014.2332953.

[151] K. Nara, Y. Mishima, and T. Satoh, "Network reconfiguration for loss minimization and load balancing", in *2003 IEEE Power Engineering Society General Meeting (IEEE Cat. No.03CH37491)*, vol. 4, 2003, 2413–2418 Vol. 4. DOI: 10.1109/PES.2003.1271019.

[152] X. Huang, Y. Yan, and Y. Zhou, "Underactuated spacecraft formation reconfiguration with collision avoidance", *Acta Astronautica*, vol. 131, pp. 166–181, 2017, ISSN: 0094-5765. DOI: https://doi.org/10.1016/j.actaastro.2016.11.037. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0094576515303672.

[153] X. Zhang, H. Duan, and C. Yang, "Pigeon-inspired optimization approach to multiple uavs formation reconfiguration controller design", in *Proceedings of 2014 IEEE Chinese Guidance, Navigation and Control Conference*, 2014, pp. 2707–2712. DOI: 10.1109/CGNCC.2014.7007594.

[154] O. Chocron, U. Prieur, and L. Pino, "A validated feasibility prototype for auv reconfigurable magnetic coupling thruster", *IEEE/ASME Transactions on Mechatronics*, vol. 19, no. 2, pp. 642–650, 2014, ISSN: 1941-014X. DOI: 10.1109/TMECH.2013.2250987.

[155] Y. Gu, B. Seanor, G. Campa, M. R. Napolitano, L. Rowe, S. Gururajan, and S. Wan, "Design and flight testing evaluation of formation control laws", *IEEE Transactions on Control Systems Technology*, vol. 14, no. 6, pp. 1105–1112, 2006, ISSN: 2374-0159. DOI: 10.1109/TCST.2006.880203.

[156] A. T. Hafez, S. N. Givigi, H. M. Schwartz, S. Yousefi, and M. Iskandarani, "Real time tactic switching for multiple cooperative uavs via model predictive control", in *2015 Annual IEEE Systems Conference (SysCon) Proceedings*, 2015, pp. 432–438. DOI: 10.1109/SYSCON.2015.7116789.

[157] G. Li, H. Xu, and Y. Lin, "Application of bat algorithm based time optimal control in multi-robots formation reconfiguration", *Journal of Bionic Engineering*, vol. 15, no. 1, pp. 126–138, 2018, ISSN: 2543-2141. DOI: 10.1007/s42235-017-0010-8. [Online]. Available: https://doi.org/10.1007/s42235-017-0010-8.

[158] M. A. Kamel, X. Yu, and Y. Zhang, "Real-time optimal formation reconfiguration of multiple wheeled mobile robots based on particle swarm optimization", in *2016 12th World Congress on Intelligent Control and Automation (WCICA)*, 2016, pp. 703–708. DOI: 10.1109/WCICA.2016.7578593.

[159] Y. Jin, H. Guo, and Y. Meng, "A hierarchical gene regulatory network for adaptive multirobot pattern formation", *IEEE Transactions on Systems, Man, and Cybernetics, Part B:Cybernetics*, vol. 42, no. 3, pp. 805–816, 2012, ISSN: 1083-4419. DOI: 10.1109/TSMCB.2011.2178021.

[160] J. W. Kwon and D. Chwa, "Hierarchical formation control based on a vector field method for wheeled mobile robots", *IEEE Transactions on Robotics*, vol. 28, no. 6, pp. 1335–1345, 2012, ISSN: 1552-3098. DOI: 10.1109/TRO.2012.2206869.

[161] Z. Duan, G. Zhai, and Z. Xiang, "State consensus for hierarchical multi-agent dynamical systems with inter-layer communication time delay", *Journal of the Franklin Institute*, vol. 352, no. 3, pp. 1235–1249, 2015, ISSN: 0016-0032. DOI: https://doi.org/10.1016/j.jfranklin.2014.12.013. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0016003214003603.

[162] Y. Liu, H. Yu, P. Shi, and C.C. Lim, "Formation control and collision avoidance for a class of multi-agent systems", *Journal of the Franklin Institute*, vol. 356, no. 10, pp. 5395 –5420, 2019, ISSN: 0016-0032. DOI: https://doi.org/10.1016/j.jfranklin.2019.05.008. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0016003219303266.

[163] Y. Liu, P. Shi, and C.C. Lim, "Collision-free formation control for multi-agent systems with dynamic mapping", *IEEE Transactions on Circuits and Systems II: Express Briefs*, DOI: 10.1109/TCSII.2019.2939583, 2019. DOI: 10.1109/TCSII.2019.2939583.

[164] L. Wang, Z. Wang, Q. Han, and G. Wei, "Synchronization control for a class of discrete-time dynamical networks with packet dropouts: A coding–decoding-based approach", *IEEE Transactions on Cybernetics*, vol. 48, no. 8, pp. 2437–2448, 2018, ISSN: 2168-2275. DOI: 10.1109/TCYB.2017.2740309.

[165] L. Wang, Z. Wang, G. Wei, and F. E. Alsaadi, "Observer-based consensus control for discrete-time multiagent systems with coding–decoding communication protocol", *IEEE Transactions on Cybernetics*, vol. 49, no. 12, pp. 4335–4345, 2019, ISSN: 2168-2275. DOI: 10.1109/TCYB.2018.2863664.

[166] S. Liu, Z. Wang, G. Wei, and M. Li, "Distributed set-membership filtering for multirate systems under the round-robin scheduling over sensor networks", *IEEE Transactions on Cybernetics*, DOI:10.1109/TCYB.2018.2885653, 2019, ISSN: 2168-2275. DOI: 10.1109/TCYB.2018.2885653.

[167] H. Zhang, Z. Cheng, G. Chen, and C. Li, "Model predictive flocking control for second-order multi-agent systems with input constraints", *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 62, no. 6, pp. 1599–1606, 2015, ISSN: 1558-0806. DOI: 10.1109/TCSI.2015.2418871.

[168] H. G. Tanner, A. Jadbabaie, and G. J. Pappas, "Flocking in fixed and switching networks", *IEEE Transactions on Automatic Control*, vol. 52, no. 5, pp. 863–868, 2007, ISSN: 2334-3303. DOI: 10.1109/TAC.2007.895948.

[169] Y. Liu, H. Yu, P. Shi, and C. C. Lim, "Formation control and collision avoidance for a class of multi-agent systems", *Journal of the Franklin Institute*, vol. 356, no. 10, pp. 5395–5420, 2019.