Swarm coordination with robust control lyapunov function : formulation and experiments

Chiew, Soon Hooi

2014

Chiew, S. H. (2014). Swarm coordination with robust control lyapunov function :
formulation and experiments. Doctoral thesis, Nanyang Technological University,
Singapore.

<https://hdl.handle.net/10356/55326>

<https://doi.org/10.32657/10356/55326>

**SWARM COORDINATION WITH ROBUST
CONTROL LYAPUNOV FUNCTION:
FORMULATION AND EXPERIMENTS**

CHIEW SOON HOOI

School of Mechanical and Aerospace Engineering

**A thesis submitted to the Nanyang Technological University
in partial fulfillment of the requirement for the degree of
Doctor of Philosophy**

2014

STATEMENT OF ORIGINALITY

I hereby certify that the work embodied in this thesis is the result of original research and has not been submitted for a higher degree to any other university or institution.

Date

Chiew Soon Hooi

To

My Family

Acknowledgement

I would like to express my deepest gratitude to my supervisor Prof. Yongki Go Tiauw Hiong who has offered me an opportunity to pursue my PhD in NTU. His guidance and sharing of knowledge throughout these four years have helped me so much towards the completion of my research works. I would also like to thank Prof Daniel New Tze How for his help to take over the role as my main supervisor in the final stage my PhD studies and to guide me towards of thesis defense.

Many thanks to my colleagues and friends I have met here who have helped me with their valuable knowledge and experience in study and research, especially Dr. Zhao Weihua who has guided me in the hardware and software development of the test platform; and Mr. Wang Jian from Technical University of Munich for his knowledge sharing during his exchange here in Singapore.

Last but not least, my heartfelt gratefulness to my family for their continuous support and understanding; and to my wife Khor Wen Fei for her encouragement that really means a lot to me.

Abstract

Flocking and swarming are inspired by nature. Swarming formation involves a large number of agents or unmanned aerial vehicles (UAVs) equipped with basic sensors or payloads. Missions are completed as a result of emergent behavior as a whole, relying on local sensing and reactive behavior. The individual position of UAVs in the swarm is not as important as in a formation flight, but rather the overall cohesion, separation and alignment requirements are the more important aspects.

A decentralized scheme with behavioral approach is deemed the most feasible for swarm coordination due to its close resemblance with flocking in nature that enables an intuitive modeling of swarming behavior. The inter-agent interaction is described by artificial potential field based on topological distance. Robust control Lyapunov function (RCLF) approach is attempted to formulate the swarm control as a decentralized robust stabilization problem.

It remains as a challenge to model system uncertainties in swarming control, such as those originated from measurement, telemetry, external disturbance or the nominal plant itself. These uncertainties could pose detrimental effect to the swarm. These problems are further complicated by the number of agents that presents in a swarm.

The coordination strategy in this research addresses these challenges and has several advantages. The scalability of the swarm is achieved with decentralized architecture that allows flexibility in altering the swarm size. This enables the attrition and expansion of the number of agents. The swarm is also robust to individual agent's failure or imperfection because the malfunction of an agent or a leader will not pose

any danger to the entire group. This is because any agent can take over the leader's role should the leader fails, without informing the other agents about this change.

Simulation and experimental results are presented to show the feasibility of the approach. The experiments were first carried out with commercial quadrotor platform. In order to have more customization and flexibility on the test platform, a custom quadrotor was developed based on Open Hardware Open Software projects. The performance of the custom made quadrotor is presented to show its comparable capability with the commercial platform. An experiment with four custom made quadrotors was performed for a three dimensional scenario in the motion capture laboratory using the Vicon system.

Table of Contents

STATEMENT OF ORIGINALITY	III
Acknowledgement.....	I
Abstract.....	II
Table of Contents	IV
List of Figures.....	VII
List of Tables	X
Nomenclature	XI
Chapter 1 Introduction.....	1
1.1 Background	1
1.2 Motivation.....	6
1.3 Contributions.....	8
1.4 Thesis Outline	9
Chapter 2 Literature Review	10
2.1 Flocking and Swarming in Nature	10
2.2 Reynolds' Behavioral Model	14
2.3 Centralized and Decentralized Architecture	15
2.3.1 Centralized Architecture	17
2.3.2 Decentralized Architecture	18
2.4 Coordination Approaches	19
2.4.1 Leader-Follower Approach.....	19
2.4.2 Virtual Structure Approach.....	21
2.4.3 Behavioral Approach	23
2.4.4 Summary of Coordination Strategies.....	25
2.5 Control Strategies.....	26
2.5.1 Optimization-Based Approach.....	26

2.5.2 Graph-Based Approach.....	29
2.5.3 Artificial Potential-Based Approach.....	31
2.5.4 Summary of Control Strategies.....	36
Chapter 3 Swarming Control.....	37
3.1 Control Framework.....	37
3.2 Artificial Potential Field	39
3.3 Sensing Range and Topological Distance	43
3.4 Consensus on Heading.....	48
3.5 Connectivity	49
Chapter 4 Robust Control Lyapunov Function	51
4.1 Background.....	51
4.2 Lyapunov Framework.....	53
4.3 Robust Control Lyapunov Function (RCLF).....	54
4.4 Previous Works Related to CLF for Formation Control.....	58
4.5 System Description of a Nonlinear Robust Stabilization Problem.....	60
4.6 Robust Backstepping and Construction of RCLF.....	62
4.6.1 Strict Feedback Form.....	62
4.6.2 RCLF Construction.....	64
Chapter 5 Simulations	71
5.1 System Modeling	71
5.2 Simulation Results	75
Chapter 6 Experiments.....	82
6.1 Quadrotor Dynamics.....	82
6.2 Position Control of Individual Quadrotor	85
6.3 Laboratory Setup.....	87
6.4 Experimental Results Using Commercial Platform.....	91
6.5 Testbed Development	96

6.5.1 Introduction to the Open Source Project.....	98
6.5.2 Hardware.....	99
6.5.3 Software	104
6.5.4 Simulink Interface.....	106
6.5.5 Telemetry Delay Test.....	107
6.5.6 State Estimation of Position, Velocity and Accelerometer Bias.....	110
6.5.7 Position-Hold Accuracy.....	112
6.5.8 A Multi-Agent Testbed.....	115
6.6 Experimental Results Using Customized Platform.....	117
Chapter 7 Conclusions and Future Work.....	122
7.1 Conclusions.....	122
7.2 Future Works	124
Bibliography	125

List of Figures

Figure 1.1 A swarming scenario [1]	1
Figure 1.2 Fundamental difference in (a) a formation and (b) a swarm [4]	2
Figure 1.3 Aerodynamic advantage in V-formation of birds [5]	4
Figure 2.1 Line formation of snow geese [5]	11
Figure 2.2 Cluster formation of European starlings [5]	11
Figure 2.3 Reynolds' boids model [26]	14
Figure 2.4 Information flow of a distributed architecture [26]	16
Figure 2.5 Architecture of a behavioural decentralized controller [58]	24
Figure 2.6 Trajectories of two agents 1 & 2 in Frenet-Serret frame [84]	33
Figure 2.7 Trajectories of six agents in swarming (with cohesion)	35
Figure 2.8 Trajectories of six agents in swarming (without heading consensus)	35
Figure 3.1 Decentralized control framework of swarm	38
Figure 3.2 Vector field of the potential	39
Figure 3.3 Excessive control effort due to large repulsive potential	40
Figure 3.4 Steering control input for the six agents	41
Figure 3.5 The steering control generated with Leonard-Jones potential (for $r_0 = 1$)	41
Figure 3.6 Sigmoid potential function	42
Figure 3.7 Local interaction of agent i with its neighboring agents [2]	43
Figure 3.8 Voronoi partitions	45
Figure 3.9 Potential field based on metric distance	46
Figure 3.10 Potential field based on topological distance	47
Figure 4.1 A robust control model	52
Figure 4.2 Lyapunov function for the "hard" control law generated from quadratic RCLF [94]	56
Figure 4.3 Lyapunov function for the "soft" control law generated from flattened RCLF [94]	56

Figure 4.4 Control efforts for “hard” and “soft” control laws [94]	57
Figure 4.5 Signal flow of diagram of the system Σ	61
Figure 5.1 A sigmoid function to describe the artificial potential of local inter-agent interaction	72
Figure 5.2 The smooth function s_2 that describes the feedback control law	74
Figure 5.3 Trajectories for a swarm with six agents	76
Figure 5.4 Separation distance to the nearest agent	77
Figure 5.5 RCLF values	77
Figure 5.6 Trajectories for a swarm with six agents (with disturbance)	79
Figure 5.7 Separation distance to the nearest agent (with disturbance)	79
Figure 5.8 RCLF values (with disturbance)	80
Figure 5.9 Trajectories for a swarm with eleven agents	81
Figure 6.1 Quadrotor model [98]	82
Figure 6.2 Two-loop position controller for quadrotors [99]	86
Figure 6.3 Vicon motion capture laboratory	87
Figure 6.4 The Vicon infrared cameras	87
Figure 6.5 Marker arrangement on one of the quadrotors	88
Figure 6.6 Modeling of UAV in Vicon Tracker	88
Figure 6.7 Implementation of swarming in motion capture laboratory	89
Figure 6.8 Simulink block of the swarming algorithm	90
Figure 6.9 Trajectories of two quadrotors under pursuit scenario	92
Figure 6.10 Separation distance for scenario in Figure 6.9	93
Figure 6.11 Trajectories of three quadrotors under swarming	94
Figure 6.12 Separation distance for swarming of three quadrotors	95
Figure 6.13 Swarming of three quadrotors in the motion capture laboratory	95
Figure 6.14 Architecture of UAS	97
Figure 6.15 Autopilot top view	99

Figure 6.16 Autopilot bottom view.....	99
Figure 6.17 In-house X240 quadrotor system	101
Figure 6.18 In-house X330 quadrotor system with Vicon markers.....	103
Figure 6.19 MAVLink through UDP or UART	105
Figure 6.20 Simulink interface of the ground control station.....	106
Figure 6.21 Simulink model for telemetry performance test.....	107
Figure 6.22 Telemetry modules	108
Figure 6.23 XBee 2.4GHz (XBP24) test results	108
Figure 6.24 XBee 900MHz (XBEE-PRO 900) test results	109
Figure 6.25 F02438 Dual TTL 3DRobotics 433MHz test results	109
Figure 6.26 Accelerometer reading in NED frame.....	111
Figure 6.27 An X330 quadrotor hovering with closed position loop	113
Figure 6.28 XYZ plots for hover test.....	114
Figure 6.29 XY plots for hover test	114
Figure 6.30 A multi-agent Simulink GCS	116
Figure 6.31 Four X330 quadrotors	117
Figure 6.32 Trajectory of four quadrotors in 3D space	118
Figure 6.33 Top view of the trajectory	119
Figure 6.34 Separation distance between four quadrotors.....	119
Figure 6.35 Height changes of four quadrotors	120
Figure 6.36 Minima potential trap	121

List of Tables

Table 1.1 Attributes of swarm and formation in general	2
Table 2.1 Comparison of leader-follower, virtual structure and behavioral approaches [26].....	25
Table 5.1 Performance with different number of agents	78
Table 6.1 Quadrotor properties	84
Table 6.2 Telemetry time delay test results	109

Nomenclature

d	=	relative distance
N	=	number of agents
r	=	Euclidean distance
$\mathbf{r}$	=	position vector
r_o	=	safe or desired separation distance
u	=	input
$\mathbf{u}$	=	general input vector
v	=	speed
V	=	Robust Control Lyapunov Function
$w(\cdot)$	=	potential
$\mathbf{w}$	=	general disturbance vector
x, y	=	Cartesian coordinates in global frame
$\mathbf{x}$	=	general state vector
$\mathbf{x}(\cdot), \mathbf{y}(\cdot)$	=	unit vectors in Frenet-Serret frame
$\mathbf{z}$	=	general transformed state vector
α, β	=	weighting parameters
κ	=	steering control (curvature)
θ	=	heading
ω	=	turn rate

Subscripts

i, j, k	=	of agent i, j, k respectively
jk	=	from agent k to agent j

Superscripts

$$\cdot = d(\cdot)/dt$$

Chapter 1 Introduction

1.1 Background

Formation flight of unmanned aerial vehicles (UAVs) has drawn great interests from the research community in recent years. In the late 1980s, this was started with the research on multi-agent system in the field of mobile robotics that involved ground vehicles, which was able to carry out cooperating task. As the technology of wireless communication became matured, this field makes much more advancement in the 1990s. Specifically, formation control of flying vehicles become a focus of such research activities starting in the early 2000s.

In the context of UAV formation, swarming formation is a relatively new research area. It involves a large number of UAVs equipped with basic sensors or payloads. Mission tasks of a swarming formation are completed as a result of emergent behavior as a whole, relying on local sensing and reactive behavior. The objective of swarming is simple: to achieve a goal collectively, while avoiding obstacles, and at the same time staying together to maintain group cohesion (Figure 1.1).

Figure 1.1 A swarming scenario [1]

In general, swarming is classified as a type of flocking behavior in nature, where a large number of agents (animals, insects, fishes, etc.) interacting with their neighbors to establish a collective behavior with a common group objective [2]. The collective or emergent behavior arises from simple rules that govern the interaction between the agents. These emergent behaviors are implicit and they arise in a probabilistic manner, rather than deterministically. This opposes the deliberate behavior found in the classical formation control where there are strict criteria to follow, such as trajectory tracking and formation geometry maintenance. Table 1.1 shows a brief comparison between swarm and formation [3]. Figure 1.2 gives a good illustration of the fundamental difference between a formation and a swarm, i.e. the position of each individual agent in a swarm is not restored to the initial position before passing the obstacles; nevertheless the group stays in cohesion.

Table 1.1 Attributes of swarm and formation in general

Attributes	Swarm	Formation
Temporal	Reactive	Predictive
Interrelationship	Simple	Complex
Predictability	Probabilistic	Deterministic
Individual worth	Expendable	Critical

Figure 1.2 Fundamental difference in (a) a formation and (b) a swarm [4]

Some beneficial attributes of swarming for applications to autonomous UAV include the inherently decentralized strategy, where the absence of a central informant or supervisor will not prevent the maintenance of group behavior. This minimizes the human operator efforts since the control is implicit, and individual control of each agent is not required. Decentralized strategy also contributes to the scalability of the swarm, because addition or attrition of agents does not affect the overall group behavior. At the same time, the swarm is more tolerant to individual agent's failure or imperfection.

Formation control of UAVs was first inspired by the organized flights of birds. In biological community, they are categorized as line formation such as V-formation of Eurasian Cranes, and cluster formation such as swarming of European Starlings. In line formation, migratory birds are able to fly for long distances across continents and oceans during migration. Engineers view the V-formation as the result of the birds taking advantage of a beneficial upwash or sidewash arising from the bird flying ahead, as shown in Figure 1.3. Nevertheless, there are some critics from the biologists and aerodynamicists that the birds are in fact not flying in the optimum position with respect to the aerodynamic upwash; and the V-formation may be due to the vision scope of the birds to keep the neighbors in sight [5]. V-formation has inspired formation flight for both manned and unmanned aircrafts [6].

Figure 1.3 Aerodynamic advantage in V-formation of birds [5]

One of the main motivations for formation flight in the late nineties, for both manned and unmanned aircrafts is the better aerodynamic efficiencies for fixed-wing aircrafts formation [7]. Recently, rotorcrafts such as helicopters and quadrotors are increasingly popular in UAV formation research for their vertical take-off and landing (VTOL) ability and more flexibility in maneuvers. Contrary to fixed-wing vehicles, these vehicles do not produce the beneficial upwash to their trailing partners to yield better aerodynamic efficiencies; hence from this point of view, it is not necessary to fix their individual position in the formation. In swarming, the main objective is to achieve cohesion, separation and alignment.

Several strategies have been introduced for steering and coordinating a formation or swarm of UAVs. In this research, a decentralized behavioral-based control strategy is proposed by utilizing the artificial potential field. Robust control Lyapunov function

(RCLF) approach is attempted to formulate the swarming control as a robust stabilization problem.

The proposed strategy also enables the multi-agents system to be scalable as only local sensing is required to keep the UAVs in cohesion. A UAV only interacts with its neighbors and no global knowledge of all other UAVs is required. Therefore, expansion and attrition of the number of UAVs are flexible. The topological distance and artificial potential field approaches will be utilized to describe the agents' interaction with their neighbors.

1.2 Motivation

In the context of multi-agent coordination, swarming has several advantages over formation control.

The large number of agents in the group would increase the reliability of the overall system, enabling the system to have better survivability and more resilient to individual agent failure [8]. Besides, the agents can collaborate to survey a larger area, with a relatively simpler sensor compared to their counterparts in formation flight, bringing swarming technology to potential applications such as terrain exploration, mapping, exploring hazardous area, rescue and relief efforts of a natural disaster (earthquake, tsunami, cyclone etc.), unknown area exploration and cooperative search [9,10]. [11] also presents the collaboration between fixed-wing UAVs and quadrotors for surveillance of an area.

From the military point-of view, swarming has potentials in surveillance and reconnaissance [9], search-and-destroy mission [12] and convoy protection of ground vehicles by UAVs [13]. More military applications are described in [3]. Other applications of swarming include remote communication relay through a networking set up by the multiple agents [14], and cooperative transportation of a large or heavy object.

Due to its simplicity nature and usually small size, agents of swarming have better mobility in complex environment, or tight and confined spaces. Urban exploration with swarming is presented in [15] with a simulation of swarming at downtown Phoenix, Arizona.

Swarming can be more efficient and effective compared to a single vehicle with complex equipments, thus enhancing task performance [8]. The use of simple and less complex robot or UAV in swarming also leads to lower manufacturing and maintenance costs [3].

1.3 Contributions

This proposed research looks to contribute in the swarming coordination area. A group of swarming agents is required to achieve inter-agent coordination, namely the cohesion, separation, and alignment based on Reynolds' model. A new approach to define the local interaction based on topological distance is proposed.

This work also aims to model the swarming coordination problem as a robust stabilization problem, utilizing a robust control Lyapunov function (RCLF) approach, which has not been applied to swarming. The local interaction behavior based on artificial potential field as well as the system uncertainties and external disturbances are modeled as decentralized robust stabilization problem.

Experiments were performed with a Vicon motion capture system to demonstrate the decentralized swarming control. A custom quadrotor testbed that achieves satisfactory agility and position hold accuracy that is comparable to the commercial platform was also developed.

A three dimensional scenario for swarming was carried out and compared to the simulation result using the custom quadrotor tested in the motion capture laboratory of NTU.

1.4 Thesis Outline

This report is outlined as follows:

Chapter 1 gives a brief introduction to the background of the research, motivation and contributions.

Chapter 2 presents the literature review related to the research topic that includes the information flow architecture, coordination approaches, control strategies etc.

Chapter 3 describes the overall swarming framework.

Chapter 4 provides the insights of the Robust Control Lyapunov Function (RCLF) approach, the Lyapunov framework and robust backstepping procedures.

Chapter 5 shows the simulation results.

Chapter 6 presents the experimental results in the motion capture laboratory using the Vicon system. Testbed development is also described here.

Chapter 7 concludes this thesis with the highlights of future works.

Chapter 2 Literature Review

This chapter presents the literature review related to the research topic. First, the swarm and flocking scenario in nature is discussed. The Reynolds' boid model governs the basics of swarming is presented. Control architecture, strategies and coordination approaches are briefly discussed.

2.1 Flocking and Swarming in Nature

The formation control and swarming of UAVs were inspired by nature. Biologists are interested in birds formation as early as 1930s [16]. In the seventies, the phenomenon had also drawn the interest of scientists in diversified disciplines, such as social science, biophysics, aeronautics, computer science, and mathematics to study the underlying mechanisms of natural schooling and flocking that exists in birds, fishes, bees, etc.

Apart from the possible improved aerodynamic efficiencies, there are other intuitive purposes in natural flocking. Ants forage to search for food in a more efficient way, by depositing a substance called pheromone as an indicator to other ants that the area has been explored before [14]. The chance of detecting predators also increases [17,18].

In the biology research community, the flocking of birds is classified into two categories – line formation and cluster formation [19]. The line formation refers to a two dimensional formation that we usually observe for larger birds, especially in the migratory birds where a V-formation is formed, as shown in Figure 2.1 for snow geese. On the other hand, cluster formation refers to a large group of birds that forms

a three dimensional entity with good bonding between the individuals, but yet without any obvious shape. A typical example of a cluster formation is the swarming behavior of European starlings at twilight shown in Figure 2.2.

Figure 2.1 Line formation of snow geese [5]

Figure 2.2 Cluster formation of European starlings [5]

The cluster formation is the analogy to the engineering swarm of this research. The swarming mechanism was characterized by Reynolds in 1987 for what he called a ‘flocking’ scenario. He developed a behavioral flocking model to simulate a group of

birds by utilizing only the local sensing of each individual agent, but exhibiting an emergent behavior [20]. More discussion about Reynolds' model will be discussed in the Section 2.2. As Reynolds considered only point-mass particles in his simulation, Tu and Terzopoulos [21] improved the work with a more realistic simulation that took into consideration animal muscles modeling in fish schooling.

In the 1930s, scientists tried to understand flocking with field observation. Nichols observed in 1931 that after a flock of pigeons make a 90° turn, the head (presumably the leader) of the flock is substituted by a pigeon which was initially at the side before the turn [22]. Based on this field observation, it cannot be deduced that the flock is headed by a leader at the front-most position. Gerald [23] observed the similar phenomenon in 1943 that no bird in the flock advances more than a body's length beyond any other birds in the turning process. This suggests that the turning for all birds in the flock happens almost simultaneously, and a leader-follower scenario is unlikely. Otherwise, all trailing birds will make a gradual turn by following the leader's trajectory.

In recent decades, technology in photography enables a better characterization of this phenomenon. Pomeroy and Heppner [24] used a 3D photographic technique in 1992 to capture the flight paths of all birds, and showed that their flight paths crossed each other during turning. This also suggests that the turning is almost simultaneous for all members in the flock, and the leader of the flock does change, or there may be more than one leader in the flock.

Recently, Ballerini et al. [25] suggested that the interaction between birds in a cluster formation depends on the topological distance rather than metric distance. This means that the interaction between a pair of birds depends on the number of the birds that

exists between both of them, but not on the metric distance (the actual distance) between the two. In other words, a bird will interact the most with their immediate neighbors, but less influenced by the birds separated by their immediate neighbors. This interaction decays as the number of birds between a pair of birds increases. In addition to this observation, each bird is found to interact with a fixed number of neighbors only, around an average of six to seven.

2.2 Reynolds' Behavioral Model

Reynolds introduced three heuristic rules for flocking as a distributed behavioral model in 1987 [20]. He developed a three dimensional simulation of flocking using three rules, with an artificial creature he called “boid”. Hence, his model is also sometimes referred to as the “boids model”. The model by far contains the most fundamental rules for swarm control. The three rules are:

1. **Cohesion:** This is the characteristic that ensures all agents stay close to one another so that the entire swarm remains as a group. It also implies flock centering where all agents are gathered around the center of the flock (as shown as $\oplus$ in Figure 2.3)
2. **Separation:** The agents must keep a safe distance with their immediate neighbors, to avoid collision.
3. **Alignment:** This rule requires the agents to have a common direction of travel, as well as achieving a consensus in speed so that the entire group can move together. This rule is also called velocity matching.

Figure 2.3 illustrates the three rules of Reynolds' model.

Figure 2.3 Reynolds' boids model [26]

2.3 Centralized and Decentralized Architecture

The architecture of information flow among the agents in a formation or swarm can be classified into centralized and decentralized schemes. The decentralized scheme can be further broken down into a purely decentralized system and a distributed system.

In a centralized scheme, there exists a central unit to coordinate all agents in the formation. States from all agents are fed to the central unit for processing to produce a maneuver strategy or individual trajectories. These commands are then broadcast to individual agents.

For a purely decentralized approach, each agent reacts only to its local environment with its own set of control objectives (cohesion, separation and alignment of a swarm). This approach is promising for swarm application and will be discussed in the next paragraph.

A distributed system (or loosely speaking, a decentralized system) shares similar characteristics with a purely decentralized system, except that there exists a global goal to influence the agents' behavior, for example through a centralized informant that broadcasts tasks and goals. For the remaining of this report, the term "decentralized" will be used to denote both purely decentralized and distributed strategies. Figure 2.4 shows the information flow for a distributed system.

Figure 2.4 Information flow of a distributed architecture [26]

Distributed architecture has the capability to deal with string instabilities. The string instabilities could happen in a swarm typically consists of a large number of agents where a leader is used to guide the entire swarm. The small perturbation originates from the leader can get amplified through a chain of information flow [27].

By regarding any two adjacent agents as a sub-system, a system is string stable if the input-output gain of the sub-systems is less than unity, i.e. the disturbances are attenuated as they are transmitted down the information chain. The globally transmitted information such as the one available in a distributed system prevents the string instabilities problem because the information is not being relayed through a series of communication nodes.

2.3.1 Centralized Architecture

Centralized architecture was used in the beginning of formation flight research, for example in [28-30]. This architecture is suitable for a formation group consisting of a few agents. The architecture is prone to failure because it relies on a single leader or source of command for coordination. This is particularly critical in a dynamically changing and unstructured environment. If the centralized supervisor is a ground station, the disruption of communication between the agents and ground station may pose a danger to the formation. The centralized architecture also introduces a large computation effort since the information from all agents is processed by one single informant or supervisor. Therefore, the approach is not suitable for formation that involves a large number of agents.

2.3.2 Decentralized Architecture

Decentralized architecture is employed in some literature such as in [31-35], for both ground and aerial vehicles. In this architecture, the data fusion and control are done locally at each agent, and the emergent property as the results of local interaction leads to the desired global functionality [26]. It is a challenge to produce a deterministic outcome with a decentralized decision-making algorithm. Conflicting decisions have to be eliminated to guarantee safety, especially in the separation term defined by Reynolds to avoid collision among agents.

Nevertheless, the decentralized scheme poses several advantages over the centralized scheme that makes it particularly suitable for swarm control. Removing a central informant or decision-making center facilitates the attrition and expansion of agents and this improves scalability and robustness of swarm. The overall fault tolerance of the system is also improved [36]. Besides, each agent only interacts with the neighbors, thus minimizing the burden of communication and shortening the response time. Occasional information exchange with a central informant or ground station is only required for mission updates such as a new target destination. It is also discussed in Chapter 2.1 that the decentralized architecture resembles the nature flocking and swarming phenomena.

2.4 Coordination Approaches

There are several basic approaches to coordinate a formation or swarm in general, namely the leader-follower approach, virtual structure approach, and behavioral approach. The literature review of these approaches is discussed briefly in this section.

2.4.1 Leader-Follower Approach

In this approach, a single or multiple agents are designated as the leader(s) of the formation, and the other agents are the followers. The leader can be physically one of the agents, or a virtual leader that broadcasts the desired trajectory to the followers. This approach can be found in [7,37-39].

Fredslund and Mataric [40] devised a leader-following scheme, with a decentralized approach where each agent is only capable of local sensing, and it assumes a single nearby agent as a leader in order to develop an overall formation. Li and Chen [41] considered the sensor constraints and communication barrier in the leader-follower scheme. Sisto and Gu [42] extended the leader-follower formation control with a two-layer fuzzy logic controller – an inter-agent coordination controller for formation keeping and collision avoidance, and a higher level fuzzy coordinator at the supervisory level. Shao et al. [43] proposed a three-level hybrid control architecture to realize the centralized and decentralized cooperative control for a group of autonomous vehicles, and graph theory is employed to maintain the formation pattern. Mariottini et al. [44] introduced a leader-follower formation control with uncalibrated vision sensor, enhanced with an Unscented Kalman Filter (UKF).

The leader-follower approach has advantages in its simplicity. Group behavior can be specified by just assigning the task to a single agent (or multiples of them designated as leaders). The main disadvantages that hinder its practical use for swarm control is the vulnerability to the leader's failure and the challenging information flow from the leader to all other follower agents. Besides, there is no explicit feedback from the followers to leader, i.e. the leader is not concerned with the ability of the followers to track its desired trajectory, and failure of the followers will not alert the leader to make appropriate adjustment, and could cause the formation to break up.

2.4.2 Virtual Structure Approach

In order to reduce the control effort and communication burden of a formation when the leader-follower approach and centralized scheme are used, virtual structure is explored to reduce the formation control problem to a lower dimensional space. Virtual structure approach for formation control is used in several literatures, for example [45-49]. The entire formation pattern is regarded as a single rigid (or semi-rigid) entity in which the agents maintain a specific geometry. Tensegrity structures such as struts and cables can be used to relax the rigidity constraint in the virtual structure approach [50].

In addition to atmospheric flight, the virtual structure is also used for spacecraft formation, for example the space-based interferometers where precise relative positioning of the interferometry sensors are critical [47,48]. In [47], the leader-following scheme, behavioral and virtual structure approaches are combined. In [28], the virtual structure approach is coupled with the optimization-based approach and graph theory to achieve trajectory tracking for a formation established with triangulated elemental structures. Ge and Fua [51] also employed a virtual structure approach, but instead of representing an agent with a node (a vertex), a queue concept is used. A queue is represented by four elements – vertices, shape, capacity, and encapsulating region, thus it is more flexible and allows more variety of formation structures. This work was extended in [52] to account for a limited communication scenario.

The advantages of virtual structure are the ease to define a coordinated behavior in terms of geometrical pattern, and the feedback is naturally defined. The main drawbacks of this approach are the limited applications due to less flexible formation

and taxing constraints. Most virtual structure applications deal with a homogeneous team; otherwise, the interrelationship between different types of agents must be explicitly incorporated into the geometrical structure. It is also computationally expensive when a large number of agents are involved.

2.4.3 Behavioral Approach

The behavioral approach is suitable for swarm application because it can handle large number of agents due to its naturally decentralized property. The group collective behavior arises from a simple rule governing the interrelationship between nearby agents. The overall emergent properties of the formation are of greater interest compared to behavior of individual agents. The approach has been reported in the literature ranging from multi-UGV (unmanned ground vehicles) formation to constellations of spacecraft.

Early studies of this topic date back to the late eighties. Yeung and Bekey [53] proposed a decentralized scheme to decompose the motion planning problem of robots in dynamic environment into a global path planning problem and a local path re-planning problem. He also showed that a centralized approach is intractable. In the early nineties, Sugihara and Suzuki [54] presented a distributed approach to organize a group of agents into a desired formation pattern, such as a circle or simple polygon. Parker [55] studied the balance between global and local knowledge to control a group behavior and concluded that local control alone is not sufficient to meet the global goal of certain tasks. Brogan and Hodgins [56] took into consideration the significant dynamics of the agents, such as legged robots in developing a decentralized scheme.

In recent years, Monteiro and Bicho [57] devised a distributed control architecture using dynamical systems theory based on behavioral approach for formation navigation and obstacle avoidance. Lawton et al. [32] decomposed the formation problem into a sequence of maneuvers to achieve complex formation maneuvers and made a demonstration with ground robots. Each robot handles inter-robot interaction

with three control strategies, each of which is influenced by relative position, inter-robot damping, and actuator saturation.

The advantages of behavioral approach are less communication requirements, and a simple control strategy that enables each agent to react to its surrounding agents with explicit feedback. Therefore it is relatively easy to implement if a formation involves a large number of agents. The main drawbacks of the approach are that the group behavior cannot be explicitly defined, and the convergent and stability of the emergent behavior is difficult to analyze mathematically.

The behavioral approach is often coupled with artificial potential field for inter-agent coordination to create simplistic control laws. The architecture of a behavioral decentralized controller is illustrated in Figure 2.5.

Figure 2.5 Architecture of a behavioural decentralized controller [58]

2.4.4 Summary of Coordination Strategies

Based on the literature review on the coordination strategy, a behavioral approach is chosen here for swarming application. The choice is primarily due to its simplicity that can handle large number of agents, like what nature does. Table 2.1 summarizes the advantages and disadvantages of the leader-follower, virtual structure, and behavioral approaches.

Table 2.1 Comparison of leader-follower, virtual structure and behavioral approaches [26]

	Advantages	Disadvantages
Leader-follower	<ul style="list-style-type: none">- Group behavior is easily formulated	<ul style="list-style-type: none">- Vulnerable to leader's failure- No explicit feedback from followers to the leader
Virtual structure	<ul style="list-style-type: none">- Easy to define coordinated behavior of the formation (geometry)- Feedback is naturally defined	<ul style="list-style-type: none">- Less flexible formation- Unnecessary constraints- Computationally expensive- Limited to homogeneous team
Behavioral	<ul style="list-style-type: none">- Explicit feedback- Simple and natural control strategies- Less communication requirement	<ul style="list-style-type: none">- Group behavior cannot be explicitly defined- Emergent behavior is difficult to analyze mathematically

2.5 Control Strategies

In addition to the coordination approaches and the centralized/decentralized architecture mentioned in previous sections, the research of formation and swarm control can also be viewed from the control strategies perspective. The most common strategies are optimization-based, graph theory-based, and artificial potential field-based.

2.5.1 Optimization-Based Approach

In an optimization-based strategy, the formation global objective is specified with a cost function. For a centralized architecture, the cost function for the overall system can be generally specified by

$$J = \int_0^T L(\mathbf{q}, \mathbf{u}) dt + V(\mathbf{q}(T), \mathbf{u}(T)) \quad (1)$$

where $\mathbf{q} \in \mathcal{Q} = \mathbb{R}^n$ represents the states of the agents and $\mathbf{u} = f(\mathbf{q})$ represents a feedback control strategy. T is the terminal time which is infinity if an infinite horizon problem is considered. $L(\cdot)$ is the incremental cost for the performance function that describes the task or objective, and $V(\cdot)$ is the terminal cost. $L(\cdot)$ can be set to zero if only the final state is of interest.

The cost function requires the states and inputs of all agents to be solved at a central processing unit and this can be computationally expensive for a system with many agents. Therefore, the task can be distributed to the individual agents, and the cost function is decomposed into N sub-problems:

$$J = \sum_{i=0}^N \left\{ \int_0^T L_i(\mathbf{q}_i, \mathbf{u}_i) dt + V_i(\mathbf{q}_i(T), \mathbf{u}_i(T)) \right\} \quad (2)$$

where N is the total number of agents and i denotes the individual agent.

Besides, the constraints on the states or inputs can be incorporated into the cost function using the Lagrange multipliers method. For a heterogeneous formation, each agent can be assigned a specialized performance function in terms of the agent's role parameter γ_i [59] and Eq. (2) can be altered as

$$J = \sum_{i=0}^N \left\{ \int_0^T L_i(\mathbf{q}_i, \mathbf{u}_i, \gamma_i) dt + V_i(\mathbf{q}_i(T), \mathbf{u}_i(T), \gamma_i(T)) \right\} \quad (3)$$

In addition to the classical optimization-based approach, Guadiano et al. solved the optimization problem with evolutionary computing algorithm, namely the genetic algorithm to evolve swarm control parameter for area coverage application [12]. Evolutionary algorithm is however generally slow to converge to the global optimum and may require considerable number of iterations. This drawback can be overcome with particle swarm optimization (PSO) which is considerably faster than the evolutionary algorithm. The use of PSO has been demonstrated in [60] to simulate multiple supersonic unmanned combat air vehicle (UCAVs). In [61] PSO is used to derive the anytime algorithm to generate real-time optimal paths to deal with pop up and moving obstacles. The quality of an anytime algorithm solution increases when the available computational time increases.

Model predictive control (MPC) is another approach in the optimization-based domain. Distributed MPC is used for multi-agent system with decentralized scheme. The agents only communicate their most recent optimal control policy to the nearby agents

in order to reach an equilibrium state cooperatively [62,63]. The distributed MPC problem is solved with mixed integer linear programming (MILP). Zhao and Go extended the MPC approach for three-dimensional formation to Robust Decentralized MPC (RDMPC) to include the model uncertainties and disturbances [64].

2.5.2 Graph-Based Approach

Another approach to control a formation is by graph theory [2,28,65-68]. In [66] for example, multiple teams of mobile robots are coordinated with graph theory to carry out terrain navigation, while maintaining the desired formation and avoiding obstacles. The formation is represented with a directed graph. A directed graph G is defined with two sets, $G = (V, E)$ with set V containing the vertices, and $E \subseteq V \times V$ represents the edges. Each vertex represents an agent, and the distance and interaction between two agents is described by the edge.

An example of graph theory is to parameterize a formation as triangulated surfaces. The surface is defined with a set of faces $F \subset V \times V \times V$ [28]. The face elements satisfy

$$f_{ijk} = (v_i, v_j, v_k) \in F \rightarrow (v_i, v_j) \in E, (v_j, v_k) \in E, (v_k, v_i) \in E, \forall f_{ijk} \in F \quad (4)$$

where subscripts i, j and k denote any three adjacent vertices and $i \neq j \neq k$. v_i is an element of set V that contains the vertices and f_{ijk} is an element of set F that contains the faces.

Set $D = \{d_{ij}\}$ contains the distance between two adjacent vertices, i.e. $d_{ij} = \|\mathbf{q}_i - \mathbf{q}_j\|$ for every set of agent i and agent j . Furthermore, the area of the triangulated surface is given in set $A = \{a_{ijk}\}$ and its elements are given by

$$a_{ijk} = \frac{1}{2}(\mathbf{q}_k - \mathbf{q}_i)^T \mathbf{S}(\mathbf{q}_j - \mathbf{q}_i)^T \quad (5)$$

with $\mathbf{S} = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$. $\mathbf{q}_i$ is the state vector of the system that represents the absolute position of the agents.

The triangulated formation graph in the Euclidean space is described by $G = (V, E, D, F, A)$. If G_d denotes the desired configuration and G_o is the initial configuration of the formation, the cost function for the generalized formation control is given by

$$L(\mathbf{q}, \mathbf{u}) = \sum_{i,j} \|G_d(\mathbf{q}, \mathbf{u}) - G_o(\mathbf{q}, \mathbf{u})\|^2 + \|\mathbf{u}_i\|^2 \quad (6)$$

with the last term being the cost component that describes the penalty on the inputs. Other penalties such as the time required to converge may also be added.

The disadvantages of the graph theory approach include the explicitness of the formation that makes it less flexible. If the formation is to be reconfigured, the formation graph has to be recalculated. The identification and ordering of agents complicate the situation further and cause the overall system to be sensitive to failure and reconfiguration.

2.5.3 Artificial Potential-Based Approach

The artificial potential field usually leads to a simplistic control law and is especially useful for behavioral approach, which has been reported in some literatures involving multi-agent coordination [69-71,34,72,73]. The inter-agent coordination and obstacle avoidance are achieved by defining the attractive and repulsive potentials that would steer the agent closer to a target (goal) and away from threat (obstacle).

In the early development of artificial potential-based approach, it was used for single robot navigation. Khatib [74] employed the approach to accomplish the obstacle avoidance task for a single robot; and Rimon and Koditschek [75] derived a navigation function for a single robot with the approach. Reif and Wang [76] presented one of the earliest works on the use of artificial potential fields for multi-agent control, which he termed “social potential fields”. The work by Leonard and Fiorelli [70] combines the virtual leader scheme with artificial potentials to demonstrate flocking of multi-agent systems. Yamaguchi [77] presented a specific application of the potential-based approach where a group of robots is required to enclose a target.

In the context of swarming, Gazi and Veysel [78] combined the artificial potential fields approach with a sliding-mode controller. Many of the works deal with the two-dimensional approach, but Paul et al. [79] extended the approach to the three-dimensional case.

A common problem faced by the artificial potentials approach is the existence of local minima due to the complex potential field. Several previous works deal with this problem by modeling the potential field with a harmonic function, such as a stream function from the potential flow theory [80], and Leonnard-Jones potential that

describes the interaction between a pair of neutral atoms or molecules [2]. Gyroscopic force control law is also used to handle the local minima problem by generating a smoother trajectory, as reported in [81-83].

The general control law of this approach for the navigation of an individual agent is given by

$$u_i = \nabla V(\mathbf{q}_i) \quad (7)$$

where $V(\cdot)$ is a potential function that depends on the states of the vehicle, $\mathbf{q}_i$.

Olfati-Saber presented a simulation with such approach by defining the attractive and repulsive functions [2]. The control law is given by

$$u_i = f_i^g + f_i^d + f_i^y \quad (8)$$

with $f_i^g = -\nabla V(\cdot)$ the gradient of the potential function $V(\cdot)$; f_i^d is the velocity consensus term that functions as a damper, and f_i^y is the feedback of the group objective.

In the leader-follower scheme, the control law is given by

$$u_i = - \sum_{j \neq i}^N \nabla V_I(\|\mathbf{q}_i - \mathbf{q}_j\|) - \sum_{k \in \mathcal{K}} \nabla V_h(\|\mathbf{q}_i - \mathbf{q}_k\|) + f_{vi} \quad (9)$$

where the subscript I denotes inter-agent interaction and subscript h denotes the navigation of leaders (single or multiple) in set $\mathcal{K}$. f_{vi} represents the dissipative force based on the velocity of agent i .

In the behavioral approach, the control law is based on the interaction with the neighboring agents. An example of this control law, with the system equations of the agents described in the Frenet-Serret frame is given by [82]. The system of equations

for an individual agent k in a swarm of N agents with steering control input u_k is described by

$$\dot{\mathbf{r}}_k = \mathbf{x}_k \quad (10)$$

$$\dot{\mathbf{x}}_k = \mathbf{y}_k u_k \quad (11)$$

$$\dot{\mathbf{y}}_k = -\mathbf{x}_k u_k \quad (12)$$

where $\mathbf{r}_k$ is the trajectory of agent k ; $\mathbf{x}_k$ and $\mathbf{y}_k$ are the moving orthonormal Frenet-Serret frame attached to agent k that represent its heading. The Frenet-Serret frame describes the kinematic properties of a particle in a 3D Euclidean space. In other words, $\mathbf{x}_k$ is the unit tangent vector to the trajectory of agent k and $\mathbf{y}_k$ is the unit normal vector. The Frenet-Serret frame is illustrated in Figure 2.6 for two agents 1 and 2 in a two dimensional Euclidean space.

Figure 2.6 Trajectories of two agents 1 & 2 in Frenet-Serret frame [84]

The steering control law u_k is obtained from the summation of all inter-agent interaction of agent k with other agents j where $1 \leq j \leq N$ and $j \neq k$. Besides, agent j must also be within the sensing range of agent k . The control law u_k is expressed in Eq. (13):

$$u_k = \sum_{j \neq k} u_{jk} \quad (13)$$

The inter-agent interaction, u_{jk} is expressed as

$$u_{jk} = -\eta \left(\frac{\mathbf{r}_{jk}}{|\mathbf{r}_{jk}|} \cdot \mathbf{x}_k \right) \left(\frac{\mathbf{r}_{jk}}{|\mathbf{r}_{jk}|} \cdot \mathbf{y}_k \right) - f(|\mathbf{r}_{jk}|) \left(\frac{\mathbf{r}_{jk}}{|\mathbf{r}_{jk}|} \cdot \mathbf{y}_k \right) + \mu \mathbf{x}_j \cdot \mathbf{y}_k \quad (14)$$

where $\mathbf{r}_{jk} = \mathbf{r}_k - \mathbf{r}_j$ is the relative position vector of agent k from agent j .

The first term in Eq. (14) aligns two agents perpendicular to their common baseline $\mathbf{r}_{jk}$. The second term controls the inter-agent spacing. The artificial potential field is described with a Leonard-Jones potential, as in Eq. (15):

$$f(|\mathbf{r}_{jk}|) = \alpha \left[1 - \left(\frac{r_o}{|\mathbf{r}_{jk}|} \right)^2 \right] \quad (15)$$

where r_o defines the desired or safe distance between the agents. The last term of Eq. (14) drives the agents to a common heading. $\eta(\cdot)$, $\mu(\cdot)$ and $\alpha(\cdot)$ in Eq. (14) and Eq. (15) are arbitrary functions that specify the weighting of the corresponding terms.

In order for the swarm to carry out a mission as a group, i.e. by maintaining the cohesion of all agents with a common heading, all three terms in Eq. (14) are required. However, the alignment and heading terms can be removed if cohesion is not required. To demonstrate this, simulations for target seeking mission are illustrated in Figure 2.7 and Figure 2.8.

Figure 2.7 shows six agents carrying out the mission in a group by maintaining group cohesion at all times. If cohesion is not required, the agents would carry out the mission separately; and only the inter-agent spacing control is important to avoid collision, as shown in Figure 2.8.

Figure 2.7 Trajectories of six agents in swarming (with cohesion)

Figure 2.8 Trajectories of six agents in swarming (without heading consensus)

2.5.4 Summary of Control Strategies

The literature review reveals that for swarming application, an artificial potential-based approach is the most feasible. This is due to its simplistic control law that allows a behavioral and decentralized approach to be employed for coordination of large number of agents. The drawback of artificial potential-based approach is primarily the local minima problem that may trap the agents in a complex potential field. However, this can be solved by modeling a smooth trajectory for the agents, for example a stream function. Several types of potential field were also reported in the literature that can overcome local minima problem, as presented in Section 2.5.3. Another drawback of the potential field approach is the probabilistic nature, which means that the solution is nondeterministic. While the group behavior for the entire swarm can be planned, it is challenging to predict the trajectories for each individual agent, especially in a dynamics environment.

Optimization-based approach could be computationally expensive for a large multi-agent system. One advantage of this approach is the ability to incorporate the constraints and penalties for the system in the cost function.

While graph-based approach can be used to model large multi-agent system, by forming a proximity net connecting all adjacent neighbors; the entire group structure is less flexible due to the constraint to maintain a specific geometrical sub-structure. Nevertheless, this approach is a good choice if exact separation distance is required.

Other approaches such as the fuzzy logic and neural networks are relatively new in swarming application and could serve as an alternative or augmentation to the chosen approach in the future.

Chapter 3 Swarming Control

This chapter discusses the overall framework of swarming. The information exchange, interactive force with the local neighbors, heading consensus, and the connectivity of the network are discussed.

3.1 Control Framework

The control framework is given in Figure 3.1. The supervisor may serve in a distributed system to broadcast global goals to the swarm. In a purely decentralized scheme, the supervisor can be excluded as some informed agents (or leaders) in the swarm that are aware of the mission goal would guide the entire swarm.

The arrows in Figure 3.1 indicate the exchange of information between agent i and agent j if they are in the neighborhood of each other, i.e. $i \in \mathcal{N}(j)$ and vice versa. Heading, speed and position information are exchanged for any two neighboring agents. When there are more than two agents that are interacting, the influence from all the agents are considered to yield the resultant potential field.

The behavioral interaction of all agents in the swarm produces an emergent swarm behavior in a mission, such as maneuver as a small swarm with small separation distance to acquire a target; or loitering at large separation distance around a target, e.g. in a surveillance mission.

Figure 3.1 Decentralized control framework of swarm

3.2 Artificial Potential Field

The attractive and repulsive forces of local inter-agent interaction are described with the artificial potential function. An attractive force (positive potential) is required to bring two agents closer to each other if their separation is greater than the desired separation distance. Similarly, the repulsive force (negative potential) drives the two agents further away from each other to avoid collision. The vector field of the potential is illustrated in Figure 3.2. A sigmoid potential field, which is described with a mathematical ‘S-shape’ function is chosen to avoid the excessively large control effort that may presents when the agents are too close to each other.

Figure 3.2 Vector field of the potential

If the function that describes the potential fields is not properly defined, it may incur an unnecessarily large control effort. An example is shown in Figure 3.3 for the trajectories of six agents. It can be seen that when two agents (UAV 1 and UAV 2) move very close to each other, a large repulsive potential field is generated, as shown

in Figure 3.4. It is observed that the steering control input commanded to the two agents (UAV 1 and UAV 2) are considerably larger than the rest of the agents.

The plot of steering control input that results from the Leonard-Jones potential, as in Eq. (15) is shown in Figure 3.5. It is observed that the choice of this type of potential is not suitable because of the consequence of large control effort when the relative distance between the agents is small.

Figure 3.3 Excessive control effort due to large repulsive potential

Figure 3.4 Steering control input for the six agents

Figure 3.5 The steering control generated with Leonard-Jones potential (for $r_0 = 1$)

In order to address this issue, the sigmoid potential field, w is chosen, described by the sigmoid function given by Eq. (16):

$$w(\mathbf{r}_{jk}) = \frac{2}{1 + \exp(\|\mathbf{r}_{jk}\| - r_o)} - 1 \quad (16)$$

where $\|\mathbf{r}_{jk}\|$ is the Euclidean norm of the relative distance between any agent j and agent k , and r_o is the desired separation distance. Figure 3.6 shows the potential function with a bound of $w \in [-1,1]$. The parameter $\|\mathbf{r}_{jk}\| - r_o$ in Eq. (16) is the deviation of the agent from the desired separation distance. As this parameter goes to $\pm\infty$, the potential generated by the sigmoid function is bounded. The potential is zero when the agent achieves exact separation distance with its neighbors.

Figure 3.6 Sigmoid potential function

3.3 Sensing Range and Topological Distance

Previous works that deal with behavioral decentralized scheme define the local interactions with a sensing radius. Local interactions are assumed with any agent that falls into the sensing range of the agent i to form a localized cluster N_i , as shown in Figure 3.7.

Figure 3.7 Local interaction of agent i with its neighboring agents [2]

This approach however poses a problem in swarming if the number of agents within the sensing range is much larger. An agent i may face complication to select the neighbors to interact with.

Besides, the observation of natural flocking of birds in cluster formation by Ballerini et al. suggested that the interaction between the agents depends on the topological distance instead of a metric distance [25]. Three-dimensional reconstruction of a flock of European starlings was carried out with advance imaging technology and analyzed stochastically. It is observed that an agent would interact the most with its immediate neighbors, and less interaction exists with the agents separated by their immediate neighbors. In other words, two agents that are farther apart in a sparse flock and two agents that are nearer to each other in a dense flock (distance is quantified in the

metric unit) equally attract each other, as long as the number of agents between the two agents is the same (in topological sense). The research also found that an agent (a European starling) interacts with at most six to seven agents.

In view of this unexplored area in engineering swarm algorithm, an approach employing the Voronoi partitions is proposed to determine the immediate neighboring agents based on the topological distance, rather than the metric distance. The hypotheses from this approach are the reduced computation efforts and a more intuitive way to describe the local interactions.

A Voronoi diagram is also referred to as Dirichlet tessellations, and the cells are called Dirichlet regions, Thiessen polytopes, or Voronoi polygons. Mathematically, consider a collection of $n > 1$ agents in a convex polytope Ω , with $\mathbf{p}_i \in \mathcal{Q}$ denotes the position of the agents.

A set of regions $\mathcal{V}(P) = \{V_i, \dots, V_n\}$, $V_i \subset \mathcal{Q}$ is the Voronoi partition, generated from the set $\mathcal{P} = \{\mathbf{p}_i, \dots, \mathbf{p}_n\}$ if $V_i = \{\mathbf{q} \in \mathcal{Q} : \|\mathbf{q} - \mathbf{p}_i\| < \|\mathbf{q} - \mathbf{p}_j\|, \forall j \neq i\}$ where $\|\cdot\|$ is the Euclidean norm. If $\mathbf{p}_i$ is the (Voronoi) neighbor of $\mathbf{p}_j$ or vice versa, the Voronoi partitions V_i and V_j are adjacent and share an edge. The edge of a Voronoi partition is defined as the locus of points equi-distant to the two nearest agents. The Voronoi neighbors of $\mathbf{p}_i$ is denoted by $\mathcal{N}(i)$; and $j \in \mathcal{N}(i)$ if and only if $i \in \mathcal{N}(j)$.

Two agents are neighbors if they share a Voronoi edge. An agent would decide which agents within its sensing range to interact with based on the Voronoi diagram. Figure 3.8 illustrates an example of Voronoi partitions for a swarm of agents. Each cell contains an agent identified as the black dot.

Figure 3.8 Voronoi partitions

The partition at the center of the circle contains the agent of interest. If we assume the agent is equipped with a sensor capable of sensing radius of an arbitrary value r , it is shown that within the sensing range, non-adjacent may be included. The Voronoi partitions approach helps to identify the correct immediate neighbors (only five agents in this case) in the sense of topological distance.

The Voronoi diagram has been used in swarming control for optimization of path planning and swarm distribution, but not in defining the topological distance as described above. Lindhe et al. developed an algorithm that combines the navigation function with Voronoi partitions approach. The algorithm can achieve obstacle avoidance in a decentralized scheme, and at the same time converges to a desirable geographical distribution [85]. Only the information from the neighboring agents is required for swarm cohesion, and no centralized control is required. The agents will steer towards the centroid of the Voronoi partitions so that the desired distribution is formed.

The comparison of the resultant potential field is given in Figure 3.9 and Figure 3.10 for both metric and topological distance where the agent of interest is located at the origin. The intensity of the contour represents the magnitude of the potential field exerted on the agent of interest. The number of influential agents in this example is reduced from eleven to five. It is observed that neighboring rule based on topological distance shrinks the area where the potential field is most significant. The simulations and experiments in Chapter 5 and Chapter 6 show that this potential field is feasible for the swarming algorithm.

Figure 3.9 Potential field based on metric distance

Figure 3.10 Potential field based on topological distance

3.4 Consensus on Heading

In a two-dimensional simulation, in addition to achieving the desired separation distance, consensus on heading is also required, as specified by the “alignment” criterion in Reynolds’ boids model.

The consensus algorithm is based on the two-dimensional Frenet-Serret frame [84] that consists of tangential and normal vectors with respect to the direction of motion, given by Eq. (17) in terms of steering control (curvature):

$$\kappa_{jk} = -\alpha w(\mathbf{r}_{jk}) \left(\frac{\mathbf{r}_{jk}}{|\mathbf{r}_{jk}|} \cdot \mathbf{y}_k \right) + \beta \mathbf{x}_j \cdot \mathbf{y}_k \quad (17)$$

with $w(|\mathbf{r}_{jk}|)$ the potential given by Eq. (16). In the three-dimensional case, an additional binormal component has to be taken into account.

The vector $\mathbf{x}_j$ and $\mathbf{y}_k$ are as illustrated in Figure 2.6. The first term in Eq. (17) directs agent j and agent k towards each other if their separation is larger than the desired distance r_o and vice versa. The second term aligns the agent to a common heading.

3.5 Connectivity

Connectivity between agents is required to guarantee any two agents that are initially connected to remain in communication. For any two agents specified by $\mathbf{x}_j$ and $\mathbf{x}_k$, the distance between them is given by

$$r_{jk} = \|\mathbf{x}_j - \mathbf{x}_k\| = \sqrt{(\mathbf{x}_j - \mathbf{x}_k)^T (\mathbf{x}_j - \mathbf{x}_k)} \quad (18)$$

For the two agents to remain connected, the following inequality for the time derivative of the square of r_{jk} is considered.

$$\frac{d(r_{jk}^2)}{dt} = 2r_{jk}\dot{r}_{jk} = 2(\mathbf{x}_j - \mathbf{x}_k)^T (\dot{\mathbf{x}}_j - \dot{\mathbf{x}}_k) \leq 0 \quad (19)$$

The potential function acting between the two agents acts to reduce r_{jk} if the agents are far apart. If the contribution of all agents within the neighborhood is considered, i.e. $k \in \mathcal{N}$, this can be described as

$$\dot{\mathbf{x}}_j = - \sum_{k \in \mathcal{N}} (\mathbf{x}_j - \mathbf{x}_k) \quad (20)$$

where $\mathcal{N}$ denotes all agents in the neighborhood of j including agent j itself.

Considering any two agents j and k , it follows from Eq. (5) that $\dot{\mathbf{x}}_j = -N\mathbf{x}_j + \sum_{k \in \mathcal{N}} \mathbf{x}_k$ and $\dot{\mathbf{x}}_k = -N\mathbf{x}_k + \sum_{j \in \mathcal{N}} \mathbf{x}_j$. Substitute these into Eq. (4), and because $\sum_{j \in \mathcal{N}} \mathbf{x}_j = \sum_{k \in \mathcal{N}} \mathbf{x}_k$ since the sets include the agent of interest itself, it can be shown that

$$\begin{aligned}
\frac{d(r_{jk}^2)}{dt} &= 2r_{jk} \left[-N\mathbf{x}_j + \sum_{k \in \mathcal{N}} \mathbf{x}_k - \left(-N\mathbf{x}_k + \sum_{j \in \mathcal{N}} \mathbf{x}_j \right) \right] \\
&= 2r_{jk} [-N(\mathbf{x}_j - \mathbf{x}_k)] = -2Nr_{jk}^2 \leq 0
\end{aligned} \tag{21}$$

where N is the number of agents in the neighborhood.

Eq. (21) provides the sufficient condition for a swarm consisting of N agents to remain connected in any future time. For the neighboring rules based on topological distance, this holds as long as the communication link remains.

Chapter 4 Robust Control Lyapunov Function

4.1 Background

In a feedback control system, the main objective is to improve its stability or to achieve command following by minimizing the error between the commanded input and the feedback signal. However, inappropriately designed feedback controller may be susceptible to uncertainties or disturbances and thus jeopardizes the stability property. The effect of uncertainties can be described by the robustness property, where it studies the functionality of a system under adverse conditions. A nominal system modeled without uncertainties may not adequately describe the system in reality, thus a robust control design is required to handle the uncertainties in the actual system as close as possible.

In a robust controller design, a system model is represented by three main components – a nominal plant G , a controller K , and uncertainties Δ , as shown in Figure 4.1. The nominal plant G is assumed to be well understood and can be modeled accurately. The nominal plant is augmented with uncertainties or disturbances input $\mathbf{v}$ to reflect how the plant would behave in reality. $\mathbf{z}$ is the penalized output that generates the disturbance input $\mathbf{v}$. A controller is to be designed to guarantee the closed-loop stability and the desired performance, subject to a set of admissible uncertainties contained in set $\mathcal{F}_\Delta$.

Figure 4.1 A robust control model

Most actual systems are nonlinear. A usual approach to pursue control design of the nonlinear system is to linearize the system about an equilibrium condition, and supplement the system with uncertainties. In this case, the set $\mathcal{F}_\Delta$ may have to be chosen large enough to accommodate the nonlinear phenomenon to compensate the restriction of plant G that is modeled as linear system. This may render the controller to be too conservative when the nonlinearities are significant and create unnecessarily large control inputs. In order to overcome this drawback, one can model both the nominal plant G and the nonlinearity contained in the uncertainties Δ to be nonlinear, and this brings to robust nonlinear control design.

4.2 Lyapunov Framework

Lyapunov function and state-space method are the general method to study nonlinear systems. The earliest framework for robust nonlinear control is the Lyapunov min-max approach or the guaranteed stability [86,87]; and the game theory and the theory of dissipativity led to the “nonlinear H_∞ ” approach [88], and input-to-state stability (ISS) relates the input-output methodology with the state-space Lyapunov stability [89].

The existence of a Lyapunov function is necessary and sufficient for the stability of a nonlinear system, and this condition is very commonly used in nonlinear stability analyses [90,91]. Traditionally, the Lyapunov theory was developed for systems without inputs, and thus applied to closed-loop control systems where feedback control laws have already been chosen. A candidate Lyapunov function is used to determine a feedback control by imposing a constraint to make the Lyapunov derivative negative. Extension of this methodology to systems with control inputs introduces the control Lyapunov function (CLF) approach.

A CLF for a general nonlinear control system $\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u})$ is defined as a candidate Lyapunov function $V(\mathbf{x})$, with the property that for every fixed $\mathbf{x} \neq 0$, there exists an admissible value $\mathbf{u}$ for the control such that $\nabla V(\mathbf{x}) \cdot f(\mathbf{x}, \mathbf{u}) < 0$. This means that a CLF is a candidate Lyapunov function whose Lyapunov derivative can be made negative pointwise by the choice of an appropriate control value. If $f(\mathbf{x}, \mathbf{u})$ is continuous and there exists a continuous state feedback for the system, such that the point $\mathbf{x} = 0$ is a globally asymptotically stable equilibrium of the closed-loop system, then by standard converse Lyapunov theorems there must exist a CLF for the system. If f is affine in the control variable, then the existence of a CLF for $\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u})$ is also sufficient for stabilizability via continuous state feedback.

In summary, there exists an analogy between a candidate Lyapunov function and a CLF. The existence of a candidate Lyapunov function is necessary and sufficient for the stability of systems without inputs; and the CLF is necessary and sufficient for the systems with control inputs. CLF has its drawback as it is unable to handle two different inputs to the nominal plant G , i.e. the inputs generated by the controller K and inputs from the uncertainties Δ . Besides, CLF admits only state feedback, but does not accommodate general types of measurement feedback. Therefore, a robust control Lyapunov function (RCLF) was proposed by Freeman and Kokotović in 1996 to overcome these problems [90].

4.3 Robust Control Lyapunov Function (RCLF)

The RCLF generalizes the CLF approach mentioned above to include both control and uncertainties inputs. One of the advantages of the RCLF approach is the ability to account for system uncertainties to achieve robust stabilization, which is its main contribution in robust nonlinear control. It is valid for all robustly stabilizable systems, as well as open-loop unstable systems.

In the optimal robust stabilization problem, one is required to solve the Hamilton-Jacobi-Isaacs (HJI) equation which is a difficult task. With the RCLF approach, the construction of a cost functional is not required. The robustly stabilizing feedback can be calculated directly from the RCLF rather than from solving the HJI equation. These control laws calculated from the RCLF are called pointwise min-norm control laws that never generate a positive feedback. A positive feedback is not desirable in a control system that seeks to minimize an error as it will enlarge the error signal and cause the system to become unstable.

There are several approaches that could be used to obtain an RCLF. One of them is the Lyapunov redesign or min-max design [86]. In this approach a CLF for the nominal plant without uncertainties is first obtained. This CLF is used to derive the RCLF for the nominal system that is subject to uncertainties. This approach has a restriction, i.e. it requires all uncertainties to enter the system through the same channels as the control variables. This restriction is known as the matching condition.

The next approach is based on the nonlinear stabilization technique in adaptive nonlinear control [92]. It is done by recursively “adding an integrator” to the system, and is known as integrator backstepping or simply backstepping. Unlike the restriction of the matching condition, backstepping has a structural strict feedback condition which is much weaker, and it is always possible to have a systematic construction of an RCLF. The results of such backstepping results was first presented in [93].

The simplest robust backstepping yields a quadratic RCLF in a set of transformed coordinates. The quadratic RCLF may generate a “hard” robust control law, and creates unnecessary high local gains in the state-space. This may cause an excessive control effort, such as high-magnitude chattering for the control input. Due to the nature of recursive backstepping design, the effect of this hardening property can propagate and get amplified in the recursive process. If this problem arises, it can be overcome by imposing a “flattened” RCLF that generates a “soft” control law.

The examples of Lyapunov function for both “hard” and “soft” control laws are shown in Figure 4.2 and Figure 4.3 respectively. By penalizing the states, the RCLF can be “flattened” to avoid the high gain scenario and possible chattering problem. The corresponding control efforts are shown in Figure 4.4.

Figure 4.2 Lyapunov function for the “hard” control law generated from quadratic RCLF [94]

Figure 4.3 Lyapunov function for the “soft” control law generated from flattened RCLF [94]

Figure 4.4 Control efforts for “hard” and “soft” control laws [94]

4.4 Previous Works Related to CLF for Formation Control

The RCLF approach in formation control is not reported in the literature yet. However, there are some previous works on formation control and trajectory tracking related to CLF approach, which is the underlying basis for the RCLF approach, and will be presented in this section.

Ogren addresses the multi-agent coordination problem with the CLF approach [95]. The goal of the study is to maintain formation and regulate the formation velocity within a task completion time. A Lyapunov formation function is devised by taking a weighted sum of the parameterized CLFs for each agent. The Lyapunov formation function for a group of m vehicles is given by

$$F(s, \mathbf{x}) = \sum_{i=1}^m \beta_i V_i(s, \mathbf{x}_i) \quad (22)$$

where β_i is the weighting parameter of each agent and $V_i(s, \mathbf{x})$ is the CLF for each agent. $s(\cdot)$ can be taken intuitively as time, i.e. $s(t) = t$, or in this paper, $s(t)$ is parameterized by incorporating the error feedback into the time evolution of s . The main drawback of this approach is that the formation control is centralized into one control Lyapunov function. The states from all agents are required to determine the Lyapunov formation function. Therefore it is not suitable for a decentralized strategy which is critical for swarming.

He and Han presented an unmanned helicopter formation control with decentralized receding horizon approach [96]. The Formation Control Lyapunov Function (FCLF) is used to maintain the formation. The full dynamics of the unmanned helicopters are taken into consideration. The leader-follower coordination approach is used, where

two followers are required to track and maintain a relative distance with the leader helicopter.

The derived formation CLF is similar to Ogren's work:

$$F(\mathbf{z}, \mathbf{y}_d, t) = \sum_{i=1}^N \beta_i V_i(\mathbf{z}, \mathbf{y}_d, t) \quad (23)$$

where $\mathbf{z}$ represents the state, and $\mathbf{y}_d$ the desired trajectory.

The approach employed by Ogren and He constructs the CLF for the whole formation by summing up the CLF for each individual agent. Even though the decentralized architecture can be used for inter-agent coordination, the CLF considers the states of all agents in the formation. This poses a restriction if the same approach is to be applied to swarming problem, where the computation of the CLF can become taxing.

The CLF approach is formulated such that each agent follows a predefined trajectory, specified by a virtual leader. This implies that a (virtual) leader-follower coordination scheme rather than a behavioral approach is used, and it is not suitable for swarming application.

Therefore, this thesis looks to construct the RCLF of swarming formation without involving the states of all the agents, but only taking into consideration their immediate neighbors. The decentralized behavioral scheme will be employed.

4.5 System Description of a Nonlinear Robust Stabilization Problem

A general nonlinear system is described by a differential equation

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}, \mathbf{w}, t) \quad (24)$$

$f: \mathcal{X} \times \mathcal{U} \times \mathcal{W} \times \mathbb{R} \rightarrow \mathcal{X}$ is a continuous function with $\mathbf{x} \in \mathcal{X}$ the state variable, $\mathbf{u} \in \mathcal{U}$ the control input, $\mathbf{w} \in \mathcal{W}$ the disturbance input, and $t \in \mathbb{R}$ an independent variable which is usually the time.

$\mathbf{y}: \mathcal{X} \times \mathbb{R} \rightarrow \mathcal{Y}$ is the measurement of the system. $\mathbf{y}(\cdot, t)$ is continuous for each fixed $t \in \mathbb{R}$ and $\mathbf{y}(\mathbf{x}, \cdot)$ is locally L_∞ for each fixed $\mathbf{x} \in \mathcal{X}$. The locally L_∞ condition means the function is essentially bounded in the neighborhood of every point. $\mathbf{y} \in Y(\mathbf{x}, t)$ represents the generalization of the output equation $\mathbf{y} = h(\mathbf{x}, t)$ and it can accommodate the measurement uncertainties, such as those caused by sensor or communication imperfection.

The disturbance is defined with the function $\mathbf{w}: \mathcal{X} \times \mathcal{U} \times \mathbb{R} \rightarrow \mathcal{W}$. $\mathbf{w}(\cdot, \cdot, t)$ is continuous for each fixed $t \in \mathbb{R}$ and $\mathbf{w}(\mathbf{x}, \mathbf{u}, \cdot)$ is locally L_∞ for each fixed $(\mathbf{x}, \mathbf{u}) \in \mathcal{X} \times \mathcal{U}$. The disturbance may come from external disturbances or feedback disturbances.

The control is defined by $\mathbf{u}: \mathcal{Y} \times \mathbb{R} \rightarrow \mathcal{U}$. $\mathbf{u}(\cdot, t)$ is continuous for each fixed $t \in \mathbb{R}$ and $\mathbf{u}(\mathbf{y}, \cdot)$ is locally L_∞ for each fixed $\mathbf{y} \in \mathcal{Y}$. The control constraint can be constant or dependent on the measurement $\mathbf{y}$.

If U, W, Y are the set-valued constraint for $\mathcal{U}, \mathcal{W}, \mathcal{Y}$ respectively, a system Σ is defined as

$$\Sigma = (f, U, W, Y) \quad (25)$$

The solution of system Σ is also the solution of the initial value problem

$$\begin{aligned}\dot{\mathbf{x}} &= f(\mathbf{x}, \mathbf{u}(\mathbf{y}(x, t), t), \mathbf{w}(\mathbf{x}, \mathbf{u}(\mathbf{y}(x, t), t), t), t) \\ \mathbf{x}(t_0) &= \mathbf{x}_0\end{aligned}\tag{26}$$

with a measurement $\mathbf{y}(\mathbf{x}, t)$, a disturbance $\mathbf{w}(\mathbf{x}, \mathbf{u}, t)$, a control $\mathbf{u}(\mathbf{y}, t)$, and an initial condition $(\mathbf{x}_0, t_0) \in \mathcal{X} \times \mathbb{R}$. The system $\Sigma = (f, U, W, Y)$ is time-invariant when the mappings f, U, W, Y are independent of t . Figure 4.5 shows the signal flow for a general system Σ .

Figure 4.5 Signal flow of diagram of the system Σ

4.6 Robust Backstepping and Construction of RCLF

4.6.1 Strict Feedback Form

An n^{th} -order uncertain nonlinear system can be written as

$$\dot{\mathbf{x}} = F(\mathbf{x}, \mathbf{w}) + G(\mathbf{x}, \mathbf{w})\mathbf{u} \quad (27)$$

where the functions F and G are continuous. It is assumed that all states are available for feedback, i.e. $Y(\mathbf{x}) = \{\mathbf{x}\}$ is assumed. The system also has a single unconstrained control input $U(\mathbf{x}) \equiv \mathcal{U} = \mathbb{R}$ and the disturbance is defined by a unit ball $W(\mathbf{x}) \equiv \mathcal{B}$.

The functions F and G can be written in the form of

$$F(\mathbf{x}, \mathbf{w}) = \begin{bmatrix} \phi_{11}(\mathbf{x}, \mathbf{w}) & \phi_{12}(\mathbf{x}, \mathbf{w}) & 0 & \cdots & 0 \\ \phi_{21}(\mathbf{x}, \mathbf{w}) & \phi_{22}(\mathbf{x}, \mathbf{w}) & \phi_{23}(\mathbf{x}, \mathbf{w}) & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \phi_{n-1,1}(\mathbf{x}, \mathbf{w}) & \phi_{n-1,2}(\mathbf{x}, \mathbf{w}) & \phi_{n-1,3}(\mathbf{x}, \mathbf{w}) & \cdots & \phi_{n-1,n}(\mathbf{x}, \mathbf{w}) \\ \phi_{n1}(\mathbf{x}, \mathbf{w}) & \phi_{n2}(\mathbf{x}, \mathbf{w}) & \phi_{n3}(\mathbf{x}, \mathbf{w}) & \cdots & \phi_{nn}(\mathbf{x}, \mathbf{w}) \end{bmatrix} \mathbf{x} + F(0, \mathbf{w}) \quad (28)$$

$$G(\mathbf{x}, \mathbf{w}) = \begin{bmatrix} 0 \\ \vdots \\ 0 \\ \phi_{n,n+1}(\mathbf{x}, \mathbf{w}) \end{bmatrix} \quad (29)$$

where $\phi_{i,j}(\mathbf{x}, \mathbf{w})$ is a continuous scalar function and depends only on the disturbance $\mathbf{w}$ and the states x_1 to x_i , i.e. $\phi_{ij}(\mathbf{x}, \mathbf{w}) = \phi_{ij}(x_1, \dots, x_i, \mathbf{w})$ for $1 \leq i \leq n$ and $1 \leq j \leq i + 1$. Also, $\phi_{i,i+1}(x_1, \dots, x_i, \mathbf{w}) \neq 0$ for all $x_1, \dots, x_i \in \mathbb{R}$ and $\mathbf{w} \in \mathcal{B}$ with $1 \leq i \leq n$ so that the system is controllable for each $\mathbf{w} \in \mathcal{B}$.

The structural conditions presented above are called a “strict feedback form” or a “lower triangular form”. An RCLF can always be found for a system in a strict feedback form. Therefore, the construction of RCLF begins with the transformation of

the original system into the strict feedback form that satisfies the structural conditions in Eq. (28) and Eq. (29). These steps are presented next in Section 4.6.2.

4.6.2 RCLF Construction

To construct the RCLF for a strict feedback system, the original states $\mathcal{X}$ are first mapped into the new states $\mathcal{Z}$ in a diffeomorphism with the smooth scalar functions $s_1(x_1)$, $s_2(x_1, x_2)$, $s_{n-1}(x_1, \dots, x_{n-1})$ that depends only on the states. For a system of n -th order, the new transformed states $\mathcal{Z}$ are defined as

$$\begin{aligned} z_1 &:= x_1 \\ z_2 &:= x_2 - z_1 s_1(x_1) \\ z_3 &:= x_3 - z_2 s_2(x_1, x_2) \\ z_n &:= x_n - z_{n-1} s_{n-1}(x_1, \dots, x_{n-1}) \end{aligned} \quad (30)$$

Eq. (30) can be written in matrix form as

$$\mathbf{z} := S(\mathbf{x})\mathbf{x} = \begin{bmatrix} 1 & 0 & 0 & 0 & \dots & 0 \\ -s_1 & 1 & 0 & 0 & \dots & 0 \\ s_1 s_2 & -s_2 & 1 & 0 & \dots & 0 \\ -s_1 s_2 s_3 & s_2 s_3 & -s_3 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ \pm s_1 \dots s_{n-1} & \mp s_2 \dots s_{n-1} & \pm s_3 \dots s_{n-1} & \dots & -s_{n-1} & 1 \end{bmatrix} \mathbf{x} \quad (31)$$

The inverse matrix of $S(\mathbf{x})$ is given by

$$S^{-1}(\mathbf{x}) = \begin{bmatrix} 1 & 0 & 0 & 0 & \dots & 0 \\ s_1 & 1 & 0 & 0 & \dots & 0 \\ 0 & s_2 & 1 & 0 & \dots & 0 \\ 0 & 0 & s_3 & 1 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & s_{n-1} & 1 \end{bmatrix} \quad (32)$$

The functions $s_i(\cdot)$ can be determined recursively to yield an RCLF $V(\mathbf{x}) := \mathbf{z}^T \mathbf{z}$ for the system.

The transformed coordinates are given by $\mathbf{z} = S(\mathbf{x})\mathbf{x}$, and its derivative

$$\dot{\mathbf{z}} = \begin{bmatrix} \frac{\partial S}{\partial x_1} \mathbf{x} & \frac{\partial S}{\partial x_2} \mathbf{x} & \cdots & \frac{\partial S}{\partial x_n} \mathbf{x} \end{bmatrix} \dot{\mathbf{x}} + S(\mathbf{x})\dot{\mathbf{x}} := T(\mathbf{x})\dot{\mathbf{x}} \quad (33)$$

Introducing a new variable Θ_{ij} into Eq. (33):

$$\dot{\mathbf{z}} = T(\mathbf{x})\dot{\mathbf{x}} = \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ \Theta_{21} & 1 & 0 & 0 & \cdots & 0 \\ \Theta_{31} & \Theta_{32} & 1 & 0 & \cdots & 0 \\ \Theta_{41} & \Theta_{42} & \Theta_{43} & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ \Theta_{n,1} & \Theta_{n,2} & \Theta_{n,3} & \cdots & \Theta_{n,n-1} & 1 \end{bmatrix} \dot{\mathbf{x}} \quad (34)$$

where Θ_{ij} represents any function that depends on the states x_1 to x_{i-1} , the functions

s_1 to s_{i-1} and its partial derivative $\frac{\partial S}{\partial x_i}$.

By choosing a simple state feedback control law $u = z_n s_n(\mathbf{x})$ for the robust stabilization problem, the original system equations can be written as

$$F(\mathbf{x}, \mathbf{w}) = \begin{bmatrix} \phi_{11} & \phi_{12} & 0 & \cdots & 0 & 0 \\ \phi_{21} & \phi_{22} & \phi_{23} & \cdots & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots & \vdots \\ \phi_{n-1,1} & \phi_{n-1,2} & \phi_{n-1,3} & \cdots & \phi_{n-1,n} & 0 \\ \phi_{n1} & \phi_{n2} & \phi_{n3} & \cdots & \phi_{n,n} & \phi_{n,n+1} \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots & 0 \\ s_1 & 1 & 0 & 0 & \cdots & 0 \\ 0 & s_2 & 1 & 0 & \cdots & 0 \\ 0 & 0 & s_3 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & s_{n-1} & 1 \\ 0 & 0 & 0 & \cdots & 0 & s_n \end{bmatrix} + F(0, \mathbf{w}) \quad (35)$$

This yields the new system equations in the transformed coordinates, and introducing a new variable Ψ_{ij} :

$$\dot{\mathbf{z}} = \begin{bmatrix} \phi_{11} + \phi_{12}s_1 & \phi_{12} & 0 & \cdots & 0 \\ \Psi_{21} & \Psi_{22} + \phi_{23}s_2 & \phi_{23} & \cdots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ \Psi_{n-1,1} & \Psi_{n-1,2} & \cdots & \ddots & \phi_{n-1,n} \\ \Psi_{n,1} & \Psi_{n,2} & \Psi_{n,3} & \cdots & \Psi_{n,n} + \phi_{n,n+1}s_n \end{bmatrix} \mathbf{z} + T(\mathbf{x})F(0, \mathbf{w}) \quad (36)$$

Eq. (36) can be decomposed into

$$\begin{aligned} \dot{\mathbf{z}} = & \begin{bmatrix} \phi_{11} & \phi_{12} & 0 & 0 & \cdots & 0 \\ \Psi_{21} & \Psi_{22} & \phi_{23} & 0 & \cdots & 0 \\ \Psi_{31} & \Psi_{32} & \Psi_{33} & \phi_{34} & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \ddots & \vdots \\ \Psi_{n-1,1} & \Psi_{n-1,2} & \Psi_{n-1,3} & \cdots & \Psi_{n-1,n-1} & \phi_{n-1,n} \\ \Psi_{n,1} & \Psi_{n,2} & \Psi_{n,3} & \Psi_{n,4} & \cdots & \Psi_{n,n} \end{bmatrix} \mathbf{z} \\ & + \begin{bmatrix} \phi_{12}s_1 & 0 & 0 & \cdots & 0 \\ 0 & \phi_{23}s_2 & 0 & \cdots & 0 \\ 0 & 0 & \phi_{34}s_3 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \phi_{n,n+1}s_n \end{bmatrix} \mathbf{z} \\ & + T(\mathbf{x})F(0, \mathbf{w}) \end{aligned} \quad (37)$$

Ψ_{ij} represents any function that depends on the states x_1 to x_i , the functions s_1 to s_{i-1}

and its partial derivative $\frac{\partial s}{\partial x_i}$, as well as the disturbance $\mathbf{w}$.

For simplicity, Eq. (37) is rewritten as

$$\dot{\mathbf{z}} = [A(\mathbf{x}, \mathbf{w}) + D(\mathbf{x}, \mathbf{w})]\mathbf{z} + T(\mathbf{x})F(0, \mathbf{w}) \quad (38)$$

The RCLF for the system is defined as $V := \mathbf{z}^T \mathbf{z}$. Taking the derivative of the RCLF with respect to time and performing some algebraic operations yields

$$\dot{V} = \mathbf{z}^T \dot{\mathbf{z}} + \dot{\mathbf{z}}^T \mathbf{z} \quad (39)$$

$$\dot{V} = \mathbf{z}^T [A(\mathbf{x}, \mathbf{w}) + A^T(\mathbf{x}, \mathbf{w}) + D(\mathbf{x}, \mathbf{w})]\mathbf{z} + 2F^T(0, \mathbf{w})T^T(\mathbf{x})\mathbf{z} \quad (40)$$

Young's inequality $2ab \leq a^2 + b^2$ is applied to the last term and this gives

$$2F^T(0, \mathbf{w})T^T(\mathbf{x})\mathbf{z} \leq (F(0, \mathbf{w}))^2 + (T^T(\mathbf{x})\mathbf{z})^2 \quad (41)$$

Therefore,

$$\dot{V} \leq \mathbf{z}^T [A(\mathbf{x}, \mathbf{w}) + A^T(\mathbf{x}, \mathbf{w}) + D(\mathbf{x}, \mathbf{w}) + T(\mathbf{x})T^T(\mathbf{x})]\mathbf{z} + \|F(0, \mathbf{w})\|^2 \quad (42)$$

The term $T(\mathbf{x})T^T(\mathbf{x})$ in Eq. (42) is a symmetric matrix and can be written as

$$T(\mathbf{x})T^T(\mathbf{x}) = I_{n \times n} + \begin{bmatrix} 0 & \bar{\Theta}_{12} & \bar{\Theta}_{13} & \cdots & \bar{\Theta}_{1,n} \\ & \bar{\Theta}_{22} & \bar{\Theta}_{23} & \cdots & \bar{\Theta}_{2,n} \\ & & \bar{\Theta}_{33} & \cdots & \bar{\Theta}_{3,n} \\ & Sym. & & \ddots & \vdots \\ & & & & \bar{\Theta}_{n,n} \end{bmatrix} \quad (43)$$

where $I_{n \times n}$ is an $n \times n$ identity matrix, and $\bar{\Theta}_{ij}$ has the same attributes as Θ_{ij} as in Eq. (34).

Eq. (42) and Eq. (43) are combined to yield the new expression for the RCLF derivative, $\dot{V}$:

$$\begin{aligned} \dot{V} \leq & \mathbf{z}^T \mathbf{z} + \|F(0, \mathbf{w})\|^2 + 2\mathbf{z}^T D(\mathbf{x}, \mathbf{w})\mathbf{z} \\ & + \mathbf{z}^T \begin{bmatrix} 2\phi_{11} & \bar{\Psi}_{12} & \bar{\Psi}_{13} & \cdots & \bar{\Psi}_{1,n} \\ & \bar{\Psi}_{22} & \bar{\Psi}_{23} & \cdots & \bar{\Psi}_{2,n} \\ & & \bar{\Psi}_{33} & \cdots & \bar{\Psi}_{3,n} \\ & Sym. & & \ddots & \vdots \\ & & & & \bar{\Psi}_{n,n} \end{bmatrix} \mathbf{z} \end{aligned} \quad (44)$$

where $\bar{\Psi}_{ij}$ has the same attributes as Ψ_{ij} , as in Eq. (36).

In order to prove that $V := \mathbf{z}^T \mathbf{z}$ is an RCLF for this system which is subject to a disturbance defined by a unit ball $\mathbf{w} \in \mathcal{B}$, the following condition should hold where c is an arbitrary value and $c > 1$

$$\max_{\mathbf{w} \in \mathcal{B}} \dot{V} \leq -(c - 1)\mathbf{z}^T \mathbf{z} + \max_{\mathbf{w} \in \mathcal{B}} \|F(0, \mathbf{w})\|^2 \quad (45)$$

Simplifying the expression for $\dot{V}$ in Eq. (44) by introducing a matrix $M(\mathbf{x}, \mathbf{w})$

$$\dot{V} \leq -(c - 1)\mathbf{z}^T \mathbf{z} - \mathbf{z}^T M(\mathbf{x}, \mathbf{w})\mathbf{z} + \|F(0, \mathbf{w})\|^2 \quad (46)$$

where the matrix $M(\mathbf{x}, \mathbf{w})$ is defined by

$$M(\mathbf{x}, \mathbf{w}) := \begin{bmatrix} -c - 2\phi_{11} - 2\phi_{12}s_1 & \bar{\Psi}_{12} & \bar{\Psi}_{13} & \cdots & \bar{\Psi}_{1,n} \\ & \bar{\Psi}_{22} - 2\phi_{23}s_2 & \bar{\Psi}_{23} & \cdots & \bar{\Psi}_{2,n} \\ & & \bar{\Psi}_{33} - 2\phi_{34}s_3 & \cdots & \bar{\Psi}_{3,n} \\ & \text{Sym.} & & \ddots & \vdots \\ & & & & \bar{\Psi}_{n,n} - 2\phi_{n,n+1}s_n \end{bmatrix} \quad (47)$$

The smooth functions $s_i(\cdot)$ can be derived from the condition that $M(\mathbf{x}, \mathbf{w})$ remains positive definite for all $\mathbf{x} \in \mathcal{X}$ and $\mathbf{w} \in \mathcal{B}$ so that the inequality of Eq. (46) holds. Since $M(\mathbf{x}, \mathbf{w})$ is a nonsingular real symmetric matrix, the positive definiteness of M is achieved by having a positive determinant for the upper left 1×1 corner, upper left 2×2 corner and so on, and the M itself.

In other words, the positive definiteness of M is guaranteed by having all leading principal minors of $M(\mathbf{x}, \mathbf{w})$ positive definite for all $\mathbf{x} \in \mathcal{X}$ and $\mathbf{w} \in \mathcal{B}$. This is also known as the Sylvester's criterion.

The first leading minor is given by:

$$M_1(x_1, \mathbf{w}) := -c - 2\phi_{11}(\mathbf{x}, \mathbf{w}) - 2\phi_{12}(\mathbf{x}, \mathbf{w})s_1(x_1) \quad (48)$$

This provides the condition to choose the function $s_1(x_1)$. In the strict feedback form discussed in the previous section, a conceptual control law $x_2 = x_1 s_1(x_1)$ is assumed in this step by considering the second state x_2 as a control variable, hence the word “conceptual” is used.

The second leading minor is given by:

$$M_2(x_1, x_2, \mathbf{w}) := \begin{bmatrix} M_1(x_1, \mathbf{w}) & \Psi_1 \\ \Psi_1 & \Psi_1 - 2\phi_{23}(x_1, x_2, \mathbf{w})s_2(x_1, x_2) \end{bmatrix} \quad (49)$$

The solution of $s_1(x_1)$ from the first leading minor can be used to calculate the function Ψ_1 . Therefore, the function $s_2(x_1, x_2)$ can be chosen by having a positive determinant of $M_2(x_1, x_2, w)$ for all $x_1, x_2 \in \mathbb{R}$ and $w \in \mathcal{B}$. This procedure is carried out recursively until all functions s_i are determined, and the feedback control $u(x) = z_n s_n(x)$ is naturally obtained.

Hence, by properly choosing the smooth functions $s_i(x_1, \dots, x_i)$, a positive definite matrix $M(x, w)$ can be obtained for all $x_1, \dots, x_i \in \mathbb{R}$ and $w \in \mathcal{B}$. This will ensure the RCLF derivative satisfies the condition

$$\max_{w \in \mathcal{B}} \dot{V} \leq -(c - 1) \mathbf{z}^T \mathbf{z} + \max_{w \in \mathcal{B}} \|F(0, \mathbf{w})\|^2 \quad (50)$$

and that $V := \mathbf{z}^T \mathbf{z}$ being the system's RCLF is proven. RCLF V also satisfies the small control property when $F(0, \mathbf{w}) = 0$. The small control property requires an arbitrarily small control input to be able to stabilize a system (by having the RCLF derivative $\dot{V}$ negative) when the states of the system are very close to the equilibrium condition. If the small control property is satisfied for all admissible disturbances, the system is robustly asymptotically stabilizable, that is the system converges to the equilibrium state of $\mathbf{x} = 0$.

For the case when $F(0, \mathbf{w}) \neq 0$, the residual set Ω is given by

$$\Omega = \left\{ \mathbf{x} \in \mathcal{X} : \mathbf{z}^T \mathbf{z} \leq \frac{1}{c - 1} \max_{w \in \mathcal{B}} \|F(0, \mathbf{w})\|^2 \right\} \quad (51)$$

The existence of the residual set Ω will prevent the system from converging to the equilibrium. Since the size of the residual set is dependent on the design parameter c , it can be made arbitrarily small by choosing a large value for c that satisfies the condition in Eq. (45). However, this manipulation is carried out in the transformed z -coordinates. In the original coordinates, the arbitrarily small residue is true for state x_1

because $z_1 = x_1$; but may not be true for the remaining states, as described in Eq. (31) and Eq. (32) where the transformation also involves the smooth functions $s_i(\cdot)$.

Chapter 5 Simulations

Some simulation results are shown in this chapter to demonstrate the feasibility of the proposed approach. A decentralized and artificial potential-based scheme is devised with the RCLF scheme for control of swarming problem. The experimental results will be presented in Chapter 6.

5.1 System Modeling

The control law generated from the RCLF is supplemented onto the nominal system of the UAVs. In this preliminary work, a nonholonomic kinematics and simplified dynamics of the agents are described with a unicycle model and single integrator, as in Eq. (52):

$$\begin{aligned}\dot{x}_i &= v_i \cos \theta_i \\ \dot{y}_i &= v_i \sin \theta_i \\ \dot{\theta}_i &= \omega_i \\ \dot{v}_i &= u_i\end{aligned}\tag{52}$$

The control law will maintain a desired safe distance between the agents with local interaction based on the artificial potential field.

A simple system model for an agent is described by

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} w & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u\tag{53}$$

with x_1 represents the relative distance to its neighbor, x_2 is the time derivative of the relative distance, u the control input and w describes an interaction with the neighbors.

From Eq. (53), the continuous scalar functions ϕ_{ij} as defined in Eq. (28) are given by

$$\begin{aligned}
\phi_{11} &= w \\
\phi_{12} &= \phi_{23} = 1 \\
\phi_{21} &= \phi_{22} = \phi_{13} = 0
\end{aligned} \tag{54}$$

As discussed in Section 3.2, a sigmoid or limiting function is to be used in an artificial potential approach to avoid excessively large control input, that is

$$w = \frac{2}{1 + e^{-(x_1 - r_0)}} - 1 \tag{55}$$

where r_0 represents the safe or desired distance with the neighboring agents. The definition essentially creates a bound of $w \in [-1, 1]$. This function is shown in Figure 5.1.

Figure 5.1 A sigmoid function to describe the artificial potential of local inter-agent interaction

The diffeomorphism and algebraic manipulation described in Section 4.6 is applied to yield the transformed system that is suitable for RCLF construction, i.e. in strict feedback form. Eq. (53) is transformed into a new coordinates system as

$$\begin{aligned} z_1 &:= x_1 \\ z_2 &:= x_2 - z_1 s_1(x_1) \end{aligned} \quad (56)$$

The arbitrary value $c = 2$ and the smooth function $s_1(x_1) = -3$ are chosen to satisfy the condition of the first leading principal minor of matrix $M(\mathbf{x}, \mathbf{w})$ in Eq. (47) to be positive for all $\mathbf{x} \in \mathcal{X}$ and $\mathbf{w} \in \mathcal{B}$, as described in Section 4.6.2. It follows that

$$\begin{aligned} M_1(x_1, \mathbf{w}) &= -c - 2\phi_{11}(\mathbf{x}, \mathbf{w}) - 2\phi_{12}(\mathbf{x}, \mathbf{w})s_1(x_1) \\ &= 4 - 2w \end{aligned} \quad (57)$$

where $w \in [-1, 1]$.

From Eq. (37) and Eq. (38), the transformed state-space model can be expressed as

$$\dot{\mathbf{z}} = [A(\mathbf{x}, w) + D(\mathbf{x}, w)]\mathbf{z} \quad (58)$$

with $\begin{bmatrix} \phi_{11} & \phi_{12} \\ \theta_1 & \theta_1 \end{bmatrix} = \begin{bmatrix} w & 1 \\ 3(w-3) & 3 \end{bmatrix}$; and $D = \begin{bmatrix} \phi_{12}s_1 & 0 \\ 0 & \phi_{23}s_2 \end{bmatrix} = \begin{bmatrix} -3 & 0 \\ 0 & s_2 \end{bmatrix}$.

Inherent in the system is the feedback control law given by $u = s_2(w, x_1, x_2)z_2$, as in Eq. (35). This smooth function s_2 has to be chosen to satisfy the condition defined in Eq. (50). The manipulation of the system variables yields the following inequality.

$$s_2 \geq \max_{w \in [-1, 1]} \left\{ \frac{4w^2 - 48w + 97}{4w - 8} \right\} \quad (59)$$

The bound of $w \in [-1, 1]$ imposed by the sigmoid function of inter-agent interaction is applied to Eq. (59). Figure 5.2 shows the plot of this function s_2 . This function directly influences the feedback gain based on the system uncertainties (in this case,

the uncertainties is described by the potential of local interaction). If an input constraint is required, the function s_2 can be chosen accordingly as long as the inequality above is not violated.

Figure 5.2 The smooth function s_2 that describes the feedback control law

By choosing $s_2 = -10$ as not to violate inequality in Eq. (59), Eq. (58) yields the new state-space model for the system in the context of RCLF

$$\begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} w - 3 & 1 \\ 3(w - 3) & -7 \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \quad (60)$$

5.2 Simulation Results

Assumption: Each agent is equipped with sensors that can track the relative position of its neighbors, and there is no delay or data loss between any agents. (In Section 0, time delays for several telemetry modules are studied for real life application).

The swarming scenario was simulated with different number of interacting agents. Each agent was initialized randomly at certain arbitrary positions and heading. One of the agents in the swarm is an informed agent (with thicker line), who knows the objectives of the mission. This informed agent is aware of the mission, but if the leader malfunctions, the swarm itself will not collapse. In contrast, they will follow a heading achieved through consensus.

The rest of the agents are unaware of the mission objectives (such as a target to be acquired, or a trajectory to track). These uninformed agents do not know which agent is a leader. The entire swarm maneuvers and maintains cohesion solely as the results of local interaction the repulsive and attractive potentials with their immediate neighbors.

In the simulation, the desired separation distance is set at 2m, and the informed agent is required to make several turn maneuvers.

Figure 5.3 shows the trajectories for the six agents. One of the agents is deliberately started far away from the rest of the group. The circles indicate the final position of the agents, with the radius equivalent to the desired separation distance. The simulation shows all agents converge to achieve cohesion. With the informed agent leading the swarm, all agents are able to reach a predefined target collectively.

Figure 5.4 to Figure 5.5 show the separation distance and the value of RCLF respectively. Due to the complex potential field among the agents, they are able to approach the desired separation distance but do not converge to the exact desired value. This is also the probabilistic characteristic when behavioral model is employed. Nevertheless, the overall swarm is able to maintain cohesion, and stability of the swarm is satisfactory. By properly adjusting the desired separation value, the agents are able to perform swarming without colliding into each other, as shown in the experimental results in Chapter 6.

Figure 5.3 Trajectories for a swarm with six agents

Figure 5.4 Separation distance to the nearest agent

Figure 5.5 RCLF values

Table 5.1 summarizes the performance of the swarm algorithm when the number of interacting agents differs, when the desired separation distance is fixed at 2m.

While the separation distance for most cases did not converge to exactly 2m, they are within around 15% of deviation from the desired value. The results in Table 6.1 also show that for a typical quadrotor in the market that measures less than 1m in diameter, inter-agent collision is avoided. The quadrotor model and specification will be discussed in Chapter 6.

Table 5.1 Performance with different number of agents

Number of interacting agents	Separation distance (m)	Average deviation (%)
2	2.003	0.150
3	1.863 – 2.274	9.149
4	2.011 - 2.014	0.629
5	1.668 – 2.256	12.610
6	1.478 – 1.887	15.453
11	1.522 – 2.027	13.078

Figure 5.6 to Figure 5.8 give a similar simulation for six agents, but with uncertainties of sensor measurement taken into account to show the robustness of the swarm and the feasibility of the RCLF approach to handle disturbance originating from the sensor noise. 5% statistical error with uniform distribution for the relative distance measurement is taken into account [97].

Figure 5.6 Trajectories for a swarm with six agents (with disturbance)

Figure 5.7 Separation distance to the nearest agent (with disturbance)

Figure 5.8 RCLF values (with disturbance)

In order to demonstrate the feasibility of the approach to handle a large number of UAVs with the same approach, Figure 5.9 shows the swarming simulation for 11 agents. The computation effort for each individual agent has not increased significantly because each of them is still interacting with six to seven immediate neighbors. This shows that the proposed control framework would be suitable for a swarm of any size due to its decentralized approach that does not impose a limit on the number of agents in a swarm.

Figure 5.9 Trajectories for a swarm with eleven agents

Chapter 6 Experiments

This chapter presents the experimental setup and results. The platform used is a quadrotor which is currently very popular as a university research platform. The experiments were carried out in the motion capture laboratory equipped with the Vicon system. At the last part of this chapter, the hardware and software development of a custom quadrotor is presented.

6.1 Quadrotor Dynamics

Quadrotor is used as an agent in the multi-agent scenario. A simple illustration of the quadrotor is given in Figure 6.1 [98].

Figure 6.1 Quadrotor model [98]

The total thrust created by the four rotors is given by

$$T = f_1 + f_2 + f_3 + f_4 \quad (61)$$

The translational and rotational dynamics are given by Eq. (62) and (63) respectively.

$$\begin{pmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ g \end{pmatrix} + \frac{1}{m} \begin{pmatrix} -\sin \theta \\ \cos \theta \sin \varphi \\ \cos \theta \cos \varphi \end{pmatrix} \cdot T \quad (62)$$

where $[x, y, z]$ are the position of the quadrotor in the inertial frame. The orientation of the quadrotor is represented by the Euler angles of roll, pitch, and yaw, designated $[\varphi, \theta, \psi]$. m and g are the mass and gravitational acceleration respectively.

$$\begin{pmatrix} \ddot{\varphi} \\ \ddot{\theta} \\ \ddot{\psi} \end{pmatrix} = f(\varphi, \theta, \psi) + g(\varphi, \theta, \psi) \tau \quad (63)$$

where

$$f(\varphi, \theta, \psi) = \begin{pmatrix} \dot{\theta}\dot{\psi} \left(\frac{I_{yy} - I_{zz}}{I_{xx}} \right) - \frac{J_p}{I_{xx}} \dot{\theta}\dot{\Omega} \\ \dot{\varphi}\dot{\psi} \left(\frac{I_{zz} - I_{xx}}{I_{yy}} \right) - \frac{J_p}{I_{yy}} \dot{\varphi}\dot{\Omega} \\ \dot{\varphi}\dot{\theta} \left(\frac{I_{xx} - I_{yy}}{I_{zz}} \right) \end{pmatrix}$$

$$g(\varphi, \theta, \psi) = \begin{pmatrix} \frac{l}{I_{xx}} & 0 & 0 \\ 0 & \frac{l}{I_{yy}} & 0 \\ 0 & 0 & \frac{l}{I_{zz}} \end{pmatrix}$$

$$\tau = \begin{bmatrix} \tau_\varphi \\ \tau_\theta \\ \tau_\psi \end{bmatrix}$$

I_{xx} , I_{yy} , I_{zz} are the moment of inertia about the x , y , z axis respectively; J_p is the moment of inertia of the rotor; $\Omega = \omega_2 + \omega_4 - \omega_1 - \omega_3$.

The quadrotor to be used in the experiment was procured from Ascending Tech GmbH and has the properties as listed in Table 6.1.

Table 6.1 Quadrotor properties

Mass (kg)	0.68
Motor arm length (m)	0.17
$I_{11} = I_{22}$ (kgm ²)	0.007
I_{33} (kgm ²)	0.012
Propeller	8 × 4.5
Flight time	20 minutes

6.2 Position Control of Individual Quadrotor

This section presents the approach for position controller for a quadrotor. The position control is required for the experiment such that quadrotors would achieve good position hold accuracy before any high level planning can be implemented. The position control of the quadrotor in the experiment is achieved with a two-loop controller using dynamic inversion developed in [99]. The inner loop is the body-fixed angular rate control while the outer loop is the position control. The 3rd order time derivative of the position is replaced with the pseudo control $v = \ddot{a}$, where $\ddot{a}$ is the acceleration vector differentiated w.r.t. the inertial frame. The direct command (pitching and rolling angular rate, and change of thrust) for the position controller using the approach presented in [99] is given in Eq. (64)

$$\begin{bmatrix} \dot{q} \\ \dot{p} \\ \dot{T} \end{bmatrix} = m \begin{bmatrix} -1/T & 0 & 0 \\ 0 & 1/T & 0 \\ 0 & 0 & -1 \end{bmatrix} M_{BW} \cdot (v + \dot{d}_w) \quad (64)$$

where M_{BW} is the transformation matrix from world frame to body frame, and $\dot{d}_w$ is the aerodynamic drag divided by mass. The two-loop position controller design is illustrated in Figure 6.2.

Figure 6.2 Two-loop position controller for quadrotors [99]

6.3 Laboratory Setup

The experiment was carried out in the NTU's motion capture laboratory, equipped with eight infrared cameras, as shown in Figure 6.3. The cameras are positioned on a metal railing at 3m height, as in Figure 6.4. The camera arrangement is capable of covering a volume of about 8m (width) x 15m (length) x 3m (height).

Figure 6.3 Vicon motion capture laboratory

Figure 6.4 The Vicon infrared cameras

The Vicon system running at 100Hz feeds the position information to the quadrotors. Each quadrotor can be distinctly identified as each carries a unique arrangement of reflective markers. Figure 6.5 and Figure 6.6 show the marker arrangement on one of the quadrotors, and its corresponding modeling in the Vicon Tracker programme.

Figure 6.5 Marker arrangement on one of the quadrotors

Figure 6.6 Modeling of UAV in Vicon Tracker

Three available quadrotors in the laboratory were used in the experiment. One of the quadrotors in the swarm is an informed agent (leader), which knows the objectives of

the mission. The rest of the agents are unaware of the mission objectives (such as a target to be acquired, or a trajectory to track). In addition, the uninformed agents do not know which agent is a leader, and the swarm maneuvers and maintain cohesion solely as the results of the repulsive and attractive potentials with their neighbors.

The setup of the experiment is illustrated in Figure 6.7. The workstation processes the raw data from the eight infrared cameras (Additional eight cameras have been added in September 2013). The position information of the quadrotors are fed to individual ground control station that communicates with only one of the quadrotors. There is no information exchange between the ground control stations that do the swarming algorithm calculation. Each of them runs its algorithm independently. This setup is to demonstrate the decentralized and behavioral approaches such that the number of agents is scalable.

Figure 6.7 Implementation of swarming in motion capture laboratory

The desired separation distance is defined as 2m. In the experiment, one of the quadrotors will be switched to manual control mode, and the response of the remaining two quadrotors is studied.

Each agent takes into account the relative position of its neighbors and averages the potential caused by the neighbors to decide on the next move. Figure 6.8 shows the simulink block that runs the swarming algorithm for three quadrotors, each named TUM, NTU, and Wand. The first two quadrotors were named according to their owners, i.e. TUM (Technical University of Munich) and NTU (Nanyang Technological University) respectively. The third quadrotor was assigned to carry a Vicon wand for identification purpose. The inputs are the current position sensed by the Vicon cameras and the output is the next desired location.

Figure 6.8 Simulink block of the swarming algorithm

6.4 Experimental Results Using Commercial Platform

The quadrotors were placed at arbitrary positions in the laboratory. They would take off autonomously from the ground and would form an initial swarm and stayed aloft until one of the quadrotors moved. In the case of three quadrotors, they formed an equilateral triangle. One of the quadrotors was switched to manual control so that it could act as a leader.

The quadrotors under autonomous mode would maintain a height of 1m from the ground for safety reason.

Two scenarios are presented. The first case is to show the inter-agent collision avoidance; while the second case shows cohesion of three quadrotors.

The first case involves only two quadrotors under a pursuit scenario. One quadrotor (NTU) was to carry out pursuit on the second quadrotor (TUM). The NTU quadrotor attempted to approach the other quadrotor with manual control so that the inter-agent collision avoidance can be tested. Figure 6.9 shows the two-dimensional trajectories in a $4\text{m} \times 7\text{m}$ space.

Figure 6.9 Trajectories of two quadrotors under pursuit scenario

The scenario is better visualized with the separation distance plot shown in Figure 6.10. With the desired separation at 2m predefined in the artificial potential field, the quadrotors separation for two agents ranges from 0.9m to 2.7m, which is measured center-to-center of the quadrotors. Each quadrotor has a center-to-propeller-tip distance of 0.27m. Therefore, the minimum safe distance to avoid collision is 0.54m.

Figure 6.10 Separation distance for scenario in Figure 6.9

The next scenario is the swarming coordination of three quadrotors. The experimental results for three quadrotors in swarming coordination are shown in Figure 6.11 and Figure 6.12. The magenta triangle in Figure 6.11 represents the initial position of the quadrotors. The green triangles represent the swarm agents' positions in the subsequent time-steps, each labeled with an incremental number.

The quadrotor named 'Wand' was controlled manually and the remaining two quadrotors maintained cohesion even though they were not aware of which quadrotor actually led the way. Therefore, this approach is feasible for a leaderless swarm where the operator needs to only control any one of the agents to control the entire swarm. This is also beneficial from the robustness point of view. The elimination of the need for a distinct leader means the swarm is able to maintain cohesion in the case of individual agent failure. If the 'Wand' failed, the swarm would stay at its current formation, either holding its position or moving at a constant speed with heading hold, depending on the condition when the leader failed.

Figure 6.11 Trajectories of three quadrotors under swarming

Figure 6.12 shows that inter-agent separation was kept above the threshold of safe separation of 0.54m. Figure 6.13 shows the experiment in NTU's motion capture laboratory.

(Video link can be found at <http://www.youtube.com/watch?v=Q9iDYxBY2uo>)

Figure 6.12 Separation distance for swarming of three quadrotors

Figure 6.13 Swarming of three quadrotors in the motion capture laboratory

6.5 Testbed Development

The high cost of quadrotor procurement, and the inability to access source code of the commercial and proprietary quadrotor system that hinder us from customizing the test platform encouraged us to develop an in-house autopilot system for quadrotor based on open source project. This sub-chapter introduces the hardware and software of the autopilot system and the quadrotor.

The architecture of the Unmanned Aerial System (UAS) is as illustrated in Figure 6.14. Currently, the ground control station, autopilot, and quadrotor system itself are ready. The onboard computer-on-module is optional and required for high level processing that requires high computation power, such as vision-based navigation and Simultaneous Localizing and Mapping (SLAM) algorithm.

Ground Control Station

Computer-on-Module (Onboard)

Universal Flight Console

Figure 6.14 Architecture of UAS

6.5.1 Introduction to the Open Source Project

The testbed development is based on open source projects. The heart of the quadrotor system – the autopilot – is based on ETH’s Pixhawk project. The Pixhawk PX2 and PX4 autopilot project releases its printed circuit board (PCB) layout and the bill of materials (B.O.M) list. The one-board solution integrates a powerful ARM-Cortex M4 processor with floating point unit (FPU), MEMS sensors (accelerometer, gyro, magnetometer), pressure sensor, analog-to-digital convertor (ADC), and numerous peripherals for GPS, telemetry etc. This makes the board an ideal platform for research purpose.

The embedded system is based on the OpenPilot project. The board is running on FreeRTOS, a multi-threaded real-time operating system. Each main task of the autopilot is written as a module with a priority assignment based on its importance. For example, the manual control module that reads and decodes the remote control has the highest priority. The inter-module communication is done through a specially generated data structure called the “UAVObject”. Every module can have access to this data structure, get and set its value. For example, the “ViconPose” UAVObject contains the X, Y, Z, Roll, Pitch, and Yaw information from the Vicon motion capture system.

A lot of works have been done to customize both the hardware and software to integrate the entire UAS system. The following sections describe the system and its performance.

6.5.2 Hardware

The autopilot is able to control different types of platforms (quadrotor, fixed-wing aircraft, helicopter etc), shown in Figure 6.15 and Figure 6.16. The arrow in Figure 6.16 indicates the front of the UAV, i.e. the positive-X direction in the longitudinal axis denoted in the body frame. It is an onboard management unit for small-size UAVs.

Figure 6.15 Autopilot top view

Figure 6.16 Autopilot bottom view

It combines an autopilot and inertial measurement unit and enables the control of an aircraft using a single-board solution. Additional I/O can be easily connected via the 30-pin expansion bus. The capabilities include airspeed hold, altitude hold, turn coordination, and GPS navigation. Data logging and manual override are also supported. All feedback loop gains are user programmable. The hardware features are listed below:

- Double IMU solutions:
 1. MPU-6050 - Six-Axis (Gyro + Accelerometer) MEMS motion tracking
 2. 3D gyro, accelerometer and magnetometer, pressure sensors
 - i. 3-axis accelerometer (BMA180: $\pm 1g$ to $\pm 16g$)
 - ii. 3-axis magnetometer (HMC5883L:)
 - iii. 3-axis gyro (L2GD20: $\pm 250/\pm 500/\pm 2000$ dps)
 - iv. Pressure sensors (MS5611-01BA03: 10 cm High resolution)
- Battery voltage regulator
- Reverse polarity protection on all power inputs
- I²C, 3x UART, PPM, analog, GPS, 2x 5V GPIO, 4x PWM / Servo
- MicroSD card slot
- Expansion bus: CAN, 2x I²C, SPI, 4x analog, 2x UART, GPIOs
- USB Serial Port (Virtual COM Port / VCP) and bootloader
- 4.5V - 6 V wide supply input range (including USB power)
- Selectable 3.3V or 5V IO for UART2 and GPS ports
- 50 x 35 x 6 mm (1.38 x 1.97 x 0.24”), 8g, with 30 x 30 mm M3 mounting holes
- Sonar, laser sensor and high level computer-on-module can be connected

The quadrotor system is integrated with the autopilot, shown in Figure 6.17. This platform was assembled in-house. The frame is custom designed, and the customized

version of I²C electronic speed controller (ESC) is used. The motor-to-motor distance is 240mm. The quadrotor weighs about 360g, and is capable of lifting a payload of around 200g.

Figure 6.17 In-house X240 quadrotor system

Some features of the X240mm quadrotor:

- The quadrotor main frame is made of a carbon fibre material.
- The entire structure is very rigid to minimize vibration that could induce sensor noise.
- It has also been rigorously tested to show its crash-proof characteristics.
- The flex propellers minimize the risk of injury; and the rubber band adapters ensure the motors will not be damaged in a crash.
- These make it a suitable test platform for researchers in the control, automation or UAV technologies domain.

Despite the advantages of the X240 quadrotor, the 6-inch propeller and the high kV of the motors (1800kV) make the quadrotor noisy during flying. For that reason, a X330 quadrotor (with 330mm motor-to-motor distance) is built based on the same philosophy, as shown in Figure 6.18. The lower kV of the motors (1100kV) makes the flying much quieter. The position hold performance of the X330 is presented in Section 6.5.7. The X330 quadrotors weighs about 600g, and is capable of lifting about 400g of payload.

Figure 6.18 In-house X330 quadrotor system with Vicon markers

6.5.3 Software

The embedded system is running a real-time operating system (FreeRTOS) onboard.

The software features are listed below:

- Fully threaded RTOS design written in C - 60% idle
- Easy to develop and test new flight control algorithm (will support Matlab code generator)
- $\geq 500\text{Hz}$ actuator update rate
- 200Hz attitude estimation update
- 3D velocity and position estimator
- Full downlink telemetry (MAVlink protocol)
- Detailed system state dumps @ 200Hz to microSD card with FAT32 file system
- All math in single precision floating point

Currently the onboard attitude estimator is a Extended Kalman Filter (EKF). There is ongoing work to incorporate Sigma Point Kalman Filter (SPKF) for data fusion which is more efficient, stable and faster. The Simulink interface for UAV formation or swarming was also developed where the communication is done with XBee 2.4GHz module. This will be presented in Section 6.5.4.

The telemetry employs MAVLink 1.0 protocol. MAVLink is an open source lightweight, header-only message marshaling library for micro air vehicles. It is currently a famous protocol for the UAV community due to its low overhead, reliability and ease of expansion. MAVLink can be communicated through a UDP or UART bridge, as shown in Figure 6.19.

Figure 6.19 MAVLink through UDP or UART

Since MAVLink 1.0 is used, it is easy to communicate with the open source QGroundControl (QGC) ground control station. The features of QGC include:

- Multiple UAV support
- 2/3D aerial maps (Google Earth support) with drag-and-drop waypoints
- In-flight manipulation of waypoints and onboard parameters
- Real-time plotting of sensor and telemetry data
- Logging and plotting of sensor data
- Head-up-display

6.5.4 Simulink Interface

The Simulink interface that is ready for a formation or swarm experiment is shown in Figure 6.20. This Simulink model acts as a ground control station and communicate with the UAV through XBee. It sends position information (from Vicon motion capture system) and commands (desired position and heading - XYZH) to the UAV using the Mavlink 1.0 protocol via serial port at a baud rate of 57,600 bps. The Simulink interface also receives downlink information (such as attitude information from the UAV that could be useful for debugging purposes).

Figure 6.20 Simulink interface of the ground control station

6.5.5 Telemetry Delay Test

Several telemetry modules were tested for its performance for indoor applications using MAVLink 1.0. Several XBee modules were tested for its performance for indoor applications. The Simulink model shown in

Figure 6.21 sends a sine wave or pulse signals to the UAV and receive the returned signals to check the time delay value. Three types of modules were tested – XBee Pro Series-1 2.4GHz (XBP24), XBee 900MHz (XBEE-PRO 900) and F02438 Dual TTL 3DRobotics 433MHz, as shown in Figure 6.22.

Figure 6.21 Simulink model for telemetry performance test

(a) XBee Pro Series-1
2.4GHz (XBP24)

(b) XBee 900MHz (XBEE-
PRO 900)

(c) F02438 Dual TTL
3DRobotics 433MHz

Figure 6.22 Telemetry modules

The results obtained for the three types of telemetry modules tested using sine wave signals are shown in Figure 6.23 to Figure 6.25. The quality of the returned signals is checked from $T=3s$ onwards. The XBP24 and F02438 Dual TTL 3DRobotics 433MHz give satisfactory results, but the XBEE-PRO 900 has long delay in data transmission and is not suitable for the experiments. The results are summarized in Table 6.2.

It was decided that XBP24 be chosen as the candidate for telemetry instead of F02438 Dual TTL 3DRobotics 433MHz because of its much lighter weight. In addition, the data transmission of XBP24 is much less susceptible to wall blockage.

Figure 6.23 XBee 2.4GHz (XBP24) test results

Figure 6.24 XBee 900MHz (XBEE-PRO 900) test results

Figure 6.25 F02438 Dual TTL 3DRobotics 433MHz test results

Table 6.2 Telemetry time delay test results

Telemetry Module	Two-way time delay (s)	
	Min	Max
XBee 2.4GHz (XBP24)	0.1	0.36
XBee 900MHz (XBEE-PRO 900)	1.0	1.3
F02438 Dual TTL 3DRobotics 433MHz	0.1	0.3

6.5.6 State Estimation of Position, Velocity and Accelerometer Bias

The Vicon motion capture system provides the position and orientation of the UAV. The velocity is therefore estimated with the onboard Kalman filter. Eq. (65) and Eq. (66) show the linear system equations for the Kalman filter that estimate the position, P , velocity, V and accelerometer bias, a^b using the accelerometer reading as input.

$$X_{k+1} = \begin{bmatrix} P_{k+1} \\ V_{k+1} \\ a_{k+1}^b \end{bmatrix} = \begin{bmatrix} 1 & T & -\frac{1}{2}T^2 \\ 0 & 1 & -T \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} P_k \\ V_k \\ a_k^b \end{bmatrix} + \begin{bmatrix} \frac{1}{2}T^2 \\ T \\ 0 \end{bmatrix} a_k + w_k \quad (65)$$

$$Y_k = [1 \quad 0 \quad 0]X_k + z_k \quad (66)$$

where $w_k = \begin{bmatrix} \tilde{P}_k \\ \tilde{V}_k \\ \tilde{a}_k^b \end{bmatrix}$ is the process noise that could present in the linear system of the estimation and z_k represents the measurement noise.

It is assumed that the average value for process noise, w and measurement noise, z are zero and there is no correlation between the two. This is a valid assumption assuming white noise prevails. The process and measurement noise covariance matrices are hence defined as

$$S_w = E(w_k w_k^T) \quad (67)$$

$$S_z = E(z_k z_k^T) \quad (68)$$

The accelerometer readings from the quadrotor during hovering were recorded to obtain the mean and standard deviation of the acceleration in the NED frame (North-East-Down Frame). Figure 6.26 shows the plot of the accelerator readings.

Figure 6.26 Accelerometer reading in NED frame

The Kalman gain can be obtained from the Eq.

$$L = A\Sigma C^T (C\Sigma C^T + S_z)^{-1} \quad (69)$$

where Σ is the error covariance matrix, and can be obtained from solving the discrete-time algebraic Riccati equation [100].

It follows that the Kalman gain is $L = \begin{bmatrix} 0.3264 \\ 2.4489 \\ -8.4903 \end{bmatrix}$.

The state estimation measurement update, corrected with the position information obtained from Vicon, is as in Equation (70)

$$\hat{X}_{k+1} = \hat{X}_k + L(P - \hat{P}) \quad (70)$$

where P is the position information obtained from Vicon, and $\hat{P}$ the position estimation.

The hovering performance of the quadrotor testbed will be presented in Section 6.5.7.

6.5.7 Position-Hold Accuracy

The quadrotor system is tested in the Vicon laboratory for its hovering performance.

The set point sent to the quadrotor is $[x, y, z, \psi] = [0, 0, -1, 0]$.

This commands the quadrotor to hover at the origin of the global frame (the NED frame) at the height of 1m and at the same time keeping the heading at 0 degree (aligned with North).

The onboard heading estimated from the MEMS magnetometer is not accurate and highly affected by the electromagnetic field originated from the environment and also the motors of the quadrotor itself. On the other hand, the heading information obtained from Vicon is very accurate.

Therefore, the heading readings from Vicon and the onboard magnetometer are compared, and a new set point for heading is calculated to compensate for the error, i.e.

$$\psi_d = \psi'_d + \hat{\psi} - \psi \quad (71)$$

where ψ_d is the actual heading command effected by the quadrotor, ψ'_d is the desired heading denoted in the global frame defined by the user, $\hat{\psi}$ is the onboard heading estimation, and ψ is the heading information acquired by Vicon.

Figure 6.27 illustrates an X330 testbed hovering with a closed position loop. The accurate position and heading information from Vicon provides a good environment for testbed development.

Figure 6.27 An X330 quadrotor hovering with closed position loop

Figure 6.28 shows the performance of the X330 under hover condition at set points of $[x, y, z, \psi] = [0, 0, -1, 0]$. The accuracy of the hover test is very good with the accuracy of $\pm 1.5\text{cm}$ for X and Y axes, and $\pm 2\text{cm}$ for Z axis.

Figure 6.29 plots the X and Y position bounded within a circle of 1.5cm radius to illustrate the hovering accuracy.

The lower noise and vibration due to its rigid carbon fibre frame, the powerful processors that allow fast sensor polling at 1000Hz, attitude estimates at 200Hz, the customized I²C electronic speed controllers (ESC) and motor update rate of at least 500Hz have all contributed to its excellent performance.

Figure 6.28 XYZ plots for hover test

Figure 6.29 XY plots for hover test

6.5.8 A Multi-Agent Testbed

The previous sections describe the development of the testbed for single UAV application. The performance of the testbed, especially the position-hold accuracy in Section 6.5.7 is crucial for extension of the single UAV application towards multi-agent application.

Several challenges have been encountered for the past few months and are successfully solved. This section briefly discusses these issues. The main issue for multi-agent application lies with the telemetry. The XBee Pro Series-1 was chosen for their outstanding performance on latency and packet loss. Even though one can assign a different channel to each pair of XBee modules to minimize RF interference, it was noticed that the interference is still apparent and it greatly affects the hover accuracy when more than one quadrotor are tested concurrently.

To solve this problem, instead of using one pair of XBee modules to control each quadrotor (four modules at the ground control station (GCS) for four quadrotors), only one XBee module is used at the GCS. It is set to Broadcast Mode such that all quadrotors receive the data packets. The Simulink GCS with a single COM port (one XBee) to control four quadrotors is shown in Figure 6.30.

As the transmitter/receiver system (TX/RX) is also in the 2.4GHz frequency spectrum, the interference is reduced by linking all receivers to one single transmitter. There are also several Wi-Fi hotspots around the laboratory, but the interference from these hotspots is minimal and we did not notice any drop in the performance.

Figure 6.30 A multi-agent Simulink GCS

Another issue with multi-agent system is the bandwidth limitation. In order to broadcast control commands to all four quadrotors at 50Hz update rate, the MAVLink message has to be customized. The data type for the control commands has been changed from floating type to integer. This change allows us to control up to six quadrotors using a single COM port.

6.6 Experimental Results Using Customized Platform

In this section, experimental results will be presented for four agents, as depicted in Figure 6.31. The quadrotors are the customized platform described in Section 6.5. The reflectiveness of the Vicon markers can be clearly seen in the picture. These markers are wrapped with 3M Scotchlite 8850 reflective tape. The roundness of the markers is crucial such that it can be recognized by the Vicon cameras.

Figure 6.31 Four X330 quadrotors

In addition of maintaining a safe distance, the agents also track the height of the informed agent while performing the swarming maneuvers. Due to space limitation, the desired separation distance is set at 1.5m, and a square maneuver was performed. The informed agent is required to make three right turns and ascend at the rate of 0.1m/s. Figure 6.32 illustrates the movement of the quadrotors in 3D space.

Figure 6.32 Trajectory of four quadrotors in 3D space

Figure 6.33 shows the top view of the trajectory. The line in bold black represents the informed agent that is aware of the mission objective. The other three agents track the height of the informed agent while trying to achieve consensus on heading and speed.

The separation distance is plotted in Figure 6.34. As the second agent (UAV2) is purposely placed further away, the initial separation distance is much larger at 1.9m initially. As all agents move in cohesion, the separation distance ranges from 1.38m to 1.63m.

The height tracking is depicted in Figure 6.35. It can be clearly visualized that the informed agent made the climb first, while the other three uninformed agents followed.

Figure 6.33 Top view of the trajectory

Figure 6.34 Separation distance between four quadrotors

Figure 6.35 Height changes of four quadrotors

In fact, it is difficult to achieve scenario in Figure 6.33, due to the cramped space. Many times the minima potential trap occurs, as in Figure 6.36. In this scenario, the informed agent was programmed to fly a circle with 2m radius. The three uninformed agents initialized near the origin but are trapped at the minima potential region. It is interesting to note that the three uninformed agents are only about 0.7m away from each other. The inter-agent clearance was just 20cm, and there was no collision.

Figure 6.36 Minima potential trap

Chapter 7 Conclusions and Future Work

7.1 Conclusions

The thesis presents the original works on swarm coordination where a large number of agents is involved, with specific application on unmanned aerial vehicles (UAVs). The research consists of two parts – theoretical formulation and experiments.

Formulation of the swarm coordination with Robust Control Lyapunov Function (RCLF), non-linearity of local interactive rule (inter-agent potential) and the robustness to external disturbance (sensing noise) are studied. A novel approach to define local neighbors based on topological approach is presented. The topological distance approach has the agent interacts with its immediate neighbors, results in a less complex potential field.

A-priori information about the leader by other agents in the swarm is not required. Based on consensus on heading and speed, each agent would react only to their surroundings. Any agent can be assigned as a leader in case the previous leader malfunctions. This minimizes the human operator's effort, and increases the robustness to individual agent's failure.

The theoretical formulation is validated with indoor experiment using quadrotors as the test platform. The position information is sensed by the motion capture system (Vicon). The experiment shows that swarm cohesion is achieved, and inter-agent collision is avoided. When the collective behavior of the swarm is more important, the behavioral approach presents a fast algorithm for online multi-agent path planning that is capable of coordinating large number of agents without being limited by the

computation power. It was proven by simulations and experiments that all three criteria of Reynolds' Boid Model are achieved with the RCLF formulation.

A research platform is developed to have more freedom on customization. A Simulink ground control station is also developed for this research. The full access of the source code to the embedded system of autopilot and the Simulink ground control station provides an ideal platform for research, compared to the off-the-shelf closed source commercial platform. The platform also includes a hard real-time RTOS (FreeRTOS). Several challenges such as telemetry delay and interference has been raised and the platform has been improved in terms of hardware and software. The capability of the platform for multi-agent has been demonstrated with experiment.

7.2 Future Works

The RCLF formulation based on potential field approach poses some drawbacks. The separation distance is difficult to control especially when the number of agents is large due to the complex potential field. In addition, there is a possibility that an agent is trapped in the minima potential region. An improvement to address these issues can be studied. For example, a hybrid control strategy can be considered.

The customized quadrotor platform that is already proven to work very well with indoor motion capture system can be further improved to extend its capability to outdoor application such as GPS- and Vision-based navigation.

As the onboard GPS interface is ready, and the GPS raw data can already be retrieved, the next step is to devise an advanced data fusion algorithm (such as Sigma Point Kalman Filter) to estimate the 3D position and velocity in an outdoor environment, without relying on any motion capture cameras. Incorporation of sonar and pressure sensors for better height control would also be done to complement the height information from GPS.

Vision-based navigation which by itself is a substantial domain requires much hardware improvement. The first step is to interface the existing autopilot board with a high level processor with greater computation power (such as Gumstix, Raspberry Pi, or Beagleboard). The next step is to integrate vision-based navigation algorithm. The open source vision algorithm could be a good start.

Bibliography

1. Kim, D.H., Wang, H., Shin, S.: Decentralized control of autonomous swarm systems using artificial potential functions: Analytical design guidelines. *Journal of Intelligent and Robotic Systems: Theory and Applications* **45**(Compendex), 369-394 (2006).
2. Olfati-Saber, R.: Flocking for multi-agent dynamic systems: Algorithms and theory. *IEEE Transactions on Automatic Control* **51**(3), 401-420 (2006).
3. Clough, B.T.: UAV Swarming. So What are Those Swarms, What are the Implications, and How Do We Handle Them. In, United States 2002, p. 17p
4. Kuo-Chi, L.: Swarming UAVS Behavior Hierarchy
Multi-Robot Systems. From Swarms to Intelligent Automata Volume III. (2005)
5. Bajec, I.L., Heppner, F.H.: Organized flight in birds. *Animal Behaviour* **78**(4), 777-789 (2009).
6. Giulietti, F., Pollini, L., Innocenti, M.: Autonomous formation flight. *IEEE Control Systems Magazine* **20**(6), 34-44 (2000).
7. Pachter, M., D'Azzo, J.J., Proud, A.W.: Tight formation flight control. *Journal of Guidance, Control, and Dynamics* **24**(2), 246-254 (2001).
8. Dudek, G., Jenkin, M.R.M., Milios, E., Wilkes, D.: Taxonomy for multi-agent robotics. *Autonomous Robots* **3**(Compendex), 375-397 (1996).
9. Beard, R.W., McLain, T.W.: Multiple UAV Cooperative Search under Collision Avoidance and Limited Range Communication Constraints. In: 42nd IEEE Conference on Decision and Control, December 9, 2003 - December 12, 2003, Maui, HI, United states 2003. Proceedings of the IEEE Conference on Decision and Control, pp. 25-30. Institute of Electrical and Electronics Engineers Inc.
10. Sujit, P., Beard, R.: Multiple UAV exploration of an unknown region. *Annals of Mathematics and Artificial Intelligence* **52**(Copyright 2009, The Institution of Engineering and Technology), 335-366 (2008).
11. Jaimes, A., Kota, S., Gomez, J.: An approach to surveillance an area using swarm of fixed wing and quad-rotor unmanned aerial vehicles UAV(s). 2008 IEEE International Conference on System of Systems Engineering, SoSE 2008 (2008).
12. Gaudiano, P., Bonabeau, E., Sharge, B.: Evolving behaviors for a swarm of unmanned air vehicles. In, Pasadena, CA, United states 2005. Proceedings - 2005 IEEE Swarm Intelligence Symposium, SIS 2005, pp. 327-334. Institute of Electrical and Electronics Engineers Computer Society
13. Barnes, L., Garcia, R., Fields, M.A., Valavanis, K.: Swarm formation control utilizing ground and aerial unmanned systems. In, Nice, France 2008. 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS. Inst. of Elec. and Elec. Eng. Computer Society
14. Sabine, H., Laurent, W., Jean-Christophe, Z., Dario, F.: Ant-based swarming with positionless micro air vehicles for communication relay. *Swarm intelligence* **2**(2-4) (2008).

15. Uzol, O., Yavrucuk, I., Sezer-Uzol, N.: Panel-method-based path planning and collaborative target tracking for swarming micro air vehicles. *Journal of aircraft* **47**(Compendex), 544-550 (2010).
16. Rackham, H.: *Poiny's natural history*, vol. X: XXXII. Harvard University Press, Cambridge, Massachusetts (1933)
17. Cullen, J.M., Shaw, E., Baldwin, H.A.: Methods for measuring the three-dimensional structure of fish schools. *Animal Behaviour* **13**(4) (1965).
18. Veherencamp, S.L.: Individual, kin, and group selection. In: Marler, P., Vandenbergh, J.G. (eds.) *Handbook of behavioral neurobiology*; V.3: Social behavior and communications, vol. 3. pp. 354 - 382. Plenum, New York (1979)
19. Heppner, F.H.: Avian flight formations. *Bird-Banding* **45**, 160 - 169 (1974).
20. Reynolds, C.W.: Flocks, herds, and schools: A distributed behavioral model. *Computer Graphics (ACM)* **21**(4), 25-34 (1987).
21. Tu, X., Terzopoulos, D.: Artificial fishes: physics, locomotion, perception, behavior. Paper presented at the Proceedings of the 21st annual conference on Computer graphics and interactive techniques,
22. Nichols, J.T.: Notes on the flocking of shore birds. *The Auk* **48**(2), 181 - 185 (1931).
23. Gerard, R.W.: Synchrony in flock wheeling. *Science* **97**(2511), 160 - 161 (1943).
24. Pomeroy, H., Heppner, F.H.: Structure of turning in airborne rock dove (*Columba livia*) flocks. *The Auk* **109**, 256 - 267 (1992).
25. Ballerini, M., Calbibbo, N., Candeir, R., Cavagna, A., Cisbani, E., Giardina, I., Lecomte, V., Orlandi, A., Parisi, G., Procaccini, A., Viale, M., Zdravkovic, V.: Interaction ruling animal collective behavior depends on topological rather than metric distance: Evidence from a field study. *Proceedings of the National Academy of Sciences of the United States of America* **105**(4), 1232-1237 (2008).
26. Bil, C.: Control design for unmanned aerial vehicle swarming. In: *Proceedings of the Institution of Mechanical Engineers; Part G; Journal of aerospace engineering*, vol. 222. vol. 4, pp. 549-567. (2008)
27. Swaroop, D., Hedrick, J.K.: String stability of interconnected systems. *IEEE Transactions on Automatic Control* **41**(Compendex), 349-357 (1996).
28. Saber, R.O., Dunbar, W.B., Murray, R.M.: Cooperative Control of Multi-Vehicle Systems Using Cost Graphs and Optimization. In: 2003 American Control Conference, June 4, 2003 - June 6, 2003, Denver, CO, United states 2003. *Proceedings of the American Control Conference*, pp. 2217-2222. Institute of Electrical and Electronics Engineers Inc.
29. Egerstedt, M., Hu, X.: Formation constrained multi-agent control. *IEEE Transactions on Robotics and Automation* **17**(Compendex), 947-951 (2001).
30. Koo, T.J., Shahruz, S.M.: Formation of a group of unmanned aerial vehicles (UAVs). In: Piscataway, NJ, USA 2001. *Proceedings of the 2001 American Control Conference*. (Cat. No.01CH37148), pp. 69-74. IEEE

31. Fujibayashi, K., Murata, S., Sugawara, K., Yamamura, M.: Self-organizing formation algorithm for active elements. In: The 21st IEEE Symposium on Reliable Distributed Systems (SRDS-2002), October 13, 2002 - October 16, 2002, Suita, Japan 2002. Proceedings of the IEEE Symposium on Reliable Distributed Systems, pp. 416-421. Institute of Electrical and Electronics Engineers Computer Society
32. Lawton, J.R.T., Beard, R.W., Young, B.J.: A decentralized approach to formation maneuvers. *IEEE Transactions on Robotics and Automation* **19**(6), 933-941 (2003).
33. Stipanovic, D.M., Inalhan, G., Teo, R., Tomlin, C.J.: Decentralized overlapping control of a formation of unmanned aerial vehicles. *Automatica* **40**(8), 1285-1296 (2004).
34. Chaimowicz, L., Michael, N., Kumar, V.: Controlling swarms of robots using interpolated implicit functions. In: 2005 IEEE International Conference on Robotics and Automation, April 18, 2005 - April 22, 2005, Barcelona, Spain 2005. Proceedings - IEEE International Conference on Robotics and Automation, pp. 2487-2492. Institute of Electrical and Electronics Engineers Inc.
35. Hsieh, M.-Y.A., Loizou, S., Kumar, V.: Stabilization of multiple robots on stable orbits via local sensing. In: 2007 IEEE International Conference on Robotics and Automation, ICRA'07, April 10, 2007 - April 14, 2007, Rome, Italy 2007. Proceedings - IEEE International Conference on Robotics and Automation, pp. 2312-2317. Institute of Electrical and Electronics Engineers Inc.
36. Sycara, K.P.: Multiagent systems. *AI Magazine* **19**(Compendex), 79-92 (1998).
37. Desai, J.P., Ostrowski, J.P., Kumar, V.: Modeling and control of formations of nonholonomic mobile robots. *IEEE Transactions on Robotics and Automation* **17**(Compendex), 905-908 (2001).
38. Takahashi, H., Nishi, H., Ohnishi, K.: Autonomous decentralized control for formation of multiple mobile robots considering ability of robot. *IEEE Transactions on Industrial Electronics* **51**(Compendex), 1272-1279 (2004).
39. Consolini, L., Morbidi, F., Prattichizzo, D., Tosques, M.: A geometric characterization of leader-follower formation control. In: 2007 IEEE International Conference on Robotics and Automation, ICRA'07, April 10, 2007 - April 14, 2007, Rome, Italy 2007. Proceedings - IEEE International Conference on Robotics and Automation, pp. 2397-2402. Institute of Electrical and Electronics Engineers Inc.
40. Fredslund, J., Mataric, M.J.: A general algorithm for robot formations using local sensing: And minimal communication. *IEEE Transactions on Robotics and Automation* **18**(Compendex), 837-846 (2002).
41. Li, Y., Chen, X.: Leader-formation navigation with sensor constraints. In: ICIA 2005: 2005 International Conference on Information Acquisition, June 27, 2005 - July 3, 2005, Hong Kong, China 2005. ICIA 2005 - Proceedings of 2005 International Conference on Information Acquisition, pp. 554-559. Inst. of Elec. and Elec. Eng. Computer Society

42. Sisto, M., Gu, D.: A fuzzy leader-follower approach to formation control of multiple mobile robots. In: 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2006, October 9, 2006 - October 15, 2006, Beijing, China 2006. IEEE International Conference on Intelligent Robots and Systems, pp. 2515-2520. Institute of Electrical and Electronics Engineers Inc.
43. Shao, J., Xie, G., Wang, L.: Leader-following formation control of multiple mobile vehicles. *IET Control Theory and Applications* **1**(Compendex), 545-552 (2007).
44. Mariottini, G.L., Morbidi, F., Prattichizzo, D., Pappas, G.J., Daniilidis, K.: Leader-follower formations: Uncalibrated vision-based localization and control. In: 2007 IEEE International Conference on Robotics and Automation, ICRA'07, April 10, 2007 - April 14, 2007, Rome, Italy 2007. Proceedings - IEEE International Conference on Robotics and Automation, pp. 2403-2408. Institute of Electrical and Electronics Engineers Inc.
45. Kar-Han, T., Lewis, M.A.: Virtual structures for high-precision cooperative mobile robotic control. In: Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS '96, 4-8 Nov. 1996, New York, NY, USA 1996. Proceedings of the 1996 IEEE/RSJ International Conference on Intelligent Robots and Systems. IROS 96. Robotic Intelligence Interacting with Dynamic Worlds (Cat. No.96CH35908), pp. 132-139. IEEE
46. Lewis, M.A., Tan, K.-H.: High precision formation control of mobile robots using virtual structures. *Autonomous Robots* **4**(Compendex), 387-403 (1997).
47. Beard, R.W., Lawton, J., Hadaegh, F.Y.: A coordination architecture for spacecraft formation control. *IEEE Transactions on Control Systems Technology* **9**(Compendex), 777-790 (2001).
48. Ren, W., Beard, R.W.: A Decentralized Scheme for Spacecraft Formation Flying via the Virtual Structure Approach. In: 2003 American Control Conference, June 4, 2003 - June 6, 2003, Denver, CO, United states 2003. Proceedings of the American Control Conference, pp. 1746-1751. Institute of Electrical and Electronics Engineers Inc.
49. Li, N.H.M., Liu, H.H.T.: Formation UAV flight control using virtual structure and motion synchronization. In, Seattle, WA, United states 2008. Proceedings of the American Control Conference, pp. 1782-1787. Institute of Electrical and Electronics Engineers Inc.
50. Nabet, B., Leonard, N.E.: Shape control of a multi-agent system using tensegrity structures. In: Lagrangian and Hamiltonian Methods for Nonlinear Control 2006. 3rd IFAC Workshop, 19-21 July 2006, Berlin, Germany 2007. Lagrangian and Hamiltonian Methods for Nonlinear Control 2006. Proceedings from the 3rd IFAC Workshop, pp. 329-339. Springer-Verlag
51. Ge, S.S., Fua, C.-H.: Queues and artificial potential trenches for multirobot formations. *IEEE Transactions on Robotics* **21**(Compendex), 646-656 (2005).
52. Fua, C.-H., Ge, S.S., Do, K.D., Lim, K.-W.: Multirobot formations based on the queue-formation scheme with limited communication. *IEEE Transactions on Robotics* **23**(Compendex), 1160-1169 (2007).

53. Yeung, D., Bekey, G.A.: A decentralized approach to the motion planning problem for multiple mobile robots. In: Proceedings of the 1987 IEEE International Conference on Robotics and Automation (Cat. No.87CH2413-3), 31 March-3 April 1987, Washington, DC, USA 1987. Proceedings of the 1987 IEEE International Conference on Robotics and Automation (Cat. No.87CH2413-3), pp. 1779-1784. IEEE Comput. Soc. Press
54. Sugihara, K., Suzuki, I.: Distributed motion coordination of multiple mobile robots. In: Proceedings. 5th IEEE International Symposium on Intelligent Control 1990 (Cat. No.90TH0333-5), 5-7 Sept. 1990, Los Alamitos, CA, USA 1990. Proceedings. 5th IEEE International Symposium on Intelligent Control 1990 (Cat. No.90TH0333-5), pp. 138-143. IEEE Comput. Soc. Press
55. Parker, L.E.: Designing control laws for cooperative agent teams. In: Proceedings of 1993 IEEE International Conference on Robotics and Automation, 2-6 May 1993, Los Alamitos, CA, USA 1993. Proceedings IEEE International Conference on Robotics and Automation (Cat. No.93CH3247-4), pp. 582-587. IEEE Comput. Soc. Press
56. Brogan, D.C., Hodgins, J.K.: Group behaviors for systems with significant dynamics. *Autonomous Robots* **4**(1), 137-153 (1997).
57. Monteiro, S., Bicho, E.: A dynamical systems approach to behavior-based formation control. In: 2002 IEEE International Conference on Robotics and Automation, May 11, 2002 - May 15, 2002, Washington, DC, United states 2002. Proceedings - IEEE International Conference on Robotics and Automation, pp. 2606-2611. Institute of Electrical and Electronics Engineers Inc.
58. Kim, S., Kim, Y.: Optimum design of three-dimensional behavioural decentralized controller for UAV formation flight. *Engineering Optimization* **41**(3), 199-224 (2009).
59. Murray, R.M.: Recent research in cooperative control of multivehicle systems. *Journal of Dynamic Systems, Measurement and Control, Transactions of the ASME* **129**(Compendex), 571-583 (2007).
60. Duan, H.-b., Ma, G.-j., Luo, D.-l.: Optimal Formation Reconfiguration Control of Multiple UCAVs Using Improved Particle Swarm Optimization. *Journal of Bionic Engineering* **5**(4), 340-347 (2008).
61. Sujit, P.B., Beard, R.: Multiple UAV path planning using anytime algorithms. In: 2009 American Control Conference, ACC 2009, June 10, 2009 - June 12, 2009, St. Louis, MO, United states 2009. Proceedings of the American Control Conference, pp. 2978-2983. Institute of Electrical and Electronics Engineers Inc.
62. Dunbar, W.B., Murray, R.M.: Model predictive control of coordinated multi-vehicle formations. In: 41st IEEE Conference on Decision and Control, December 10, 2002 - December 13, 2002, Las Vegas, NV, United states 2002. Proceedings of the IEEE Conference on Decision and Control, pp. 4631-4636. Institute of Electrical and Electronics Engineers Inc.

63. Borrelli, F., Keviczky, T., Balas, G.J.: Collision-free UAV formation flight using decentralized optimization and invariant sets. In, Piscataway, NJ, USA 2004. 2004 43rd IEEE Conference on Decision and Control (CDC) (IEEE Cat. No.04CH37601), pp. 1099-1104. IEEE
64. Zhao, W., Go, T.H.: 3-D Formulation of Formation Flight Based on Model Predictive Control with Collision Avoidance Scheme. Paper presented at the 48th AIAA Aerospace Sciences Meeting Including the New Horizons Forum and Aerospace Exposition, Orlando, Florida, Jan. 4-7, 2010
65. Olfati-Saber, R., Murray, R.M.: Graph rigidity and distributed formation stabilization of multi-vehicle systems. In: Proceedings of IEEE Conference on Decision and Control, 10-13 Dec. 2002, Piscataway, NJ, USA 2002. 41st IEEE Conference on Decision and Control. Proceedings (Cat. No.02CH37375), pp. 2965-2971. IEEE
66. Desai, J.P.: A graph theoretic approach for modeling mobile robot team formations. *Journal of Robotic Systems* **19**(Compendex), 511-525 (2002).
67. Fierro, R., Das, A.K.: A modular architecture for formation control. In: Proceedings of the Third International Workshop on Robot Motion and Control. RoMoCo'02, 9-11 Nov. 2002, Poznan, Poland 2002. Proceedings of the Third International Workshop on Robot Motion and Control. RoMoCo'02 (IEEE Cat. No.02EX616), pp. 285-290. Poznan Univ. Technol
68. Yangmin, L., Xin, C.: Stability on multi-robot formation with dynamic interaction topologies. In: 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2-6 Aug. 2005, Piscataway, NJ, USA 2005. 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 394-399. IEEE
69. Balch, T., Hybinette, M.: Social potentials for scalable multi-robot formations. In: Proceedings 2000 ICRA. IEEE International Conference on Robotics and Automation, 24-28 April 2000, Piscataway, NJ, USA 2000. Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065), pp. 73-80. IEEE
70. Leonard, N.E., Fiorelli, E.: Virtual leaders, artificial potentials and coordinated control of groups. In: 40th IEEE Conference on Decision and Control (CDC), December 4, 2001 - December 7, 2001, Orlando, FL, United states 2001. Proceedings of the IEEE Conference on Decision and Control, pp. 2968-2973. Institute of Electrical and Electronics Engineers Inc.
71. Ge, S.S., Fua, C.-H., Liew, W.-M.: Swarm formations using the general formation potential function. In: 2004 IEEE Conference on Robotics, Automation and Mechatronics, December 1, 2004 - December 3, 2004, Singapore 2004. 2004 IEEE Conference on Robotics, Automation and Mechatronics, pp. 655-660. Institute of Electrical and Electronics Engineers Inc.
72. Elkaim, G.H., Kelbley, R.J.: A lightweight formation control methodology for a swarm of non-holonomic vehicles. In: 2006 IEEE Aerospace Conference, March 4, 2006 - March 11, 2006, Big Sky, MT, United states 2006. IEEE Aerospace Conference Proceedings. Inst. of Elec. and Elec. Eng. Computer Society

73. Gazi, V., Fidan, B., Hanay, Y.S., Koksall, M.I.: Aggregation, foraging, and formation control of swarms with non-holonomic agents using potential functions and sliding mode techniques. *Turkish Journal of Electrical Engineering and Computer Sciences* **15**(Compendex), 149-168 (2007).
74. Khatib, O.: Real-time obstacle avoidance for manipulators and mobile robots. *International Journal of Robotics Research* **5**(Compendex), 90-98 (1986).
75. Rimon, E., Koditschek, D.E.: Exact robot navigation using artificial potential functions. *IEEE Transactions on Robotics and Automation* **8**(Compendex), 501-518 (1992).
76. Reif, J.H., Wang, H.: Social potential fields: A distributed behavioral control for autonomous robots. *Robotics and Autonomous Systems* **27**(Compendex), 171-194 (1999).
77. Yamaguchi, H.: A cooperative hunting behavior by mobile-robot troops. *International Journal of Robotics Research* **18**(Copyright 1999, IEE), 931-940 (1999).
78. Gazi, V.: Swarm aggregations using artificial potentials and sliding-mode control. *IEEE Transactions on Robotics* **21**(Compendex), 1208-1214 (2005).
79. Paul, T., Krogstad, T.R., Gravdahl, J.T.: UAV formation flight using 3D potential field. In: Ajaccio-Corsica, France 2008. 2008 Mediterranean Conference on Control and Automation - Conference Proceedings, MED'08, pp. 1240-1245. Inst. of Elec. and Elec. Eng. Computer Society
80. Waydo, S., Murray, R.M.: Vehicle motion planning using stream functions. In: 2003 IEEE International Conference on Robotics and Automation, September 14, 2003 - September 19, 2003, Taipei, Taiwan 2003. Proceedings - IEEE International Conference on Robotics and Automation, pp. 2484-2491. Institute of Electrical and Electronics Engineers Inc.
81. Chang, D.E., Shadden, S.C., Marsden, J.E., Olfati-Saber, R.: Collision avoidance for multiple agent systems. In: Decision and Control, 2003. Proceedings. 42nd IEEE Conference on 2003, pp. 539-543 Vol.531
82. Morgan, D.S., Schwartz, I.B.: Dynamic coordinated control laws in multiple agent models. *Physics letters. A* **340**(1), 121-131 (2005).
83. Haibo, M., Fuchun, S., Feng, N.: Decentralized UAV formation tracking flight control using gyroscopic force. In: 2009 IEEE International Conference on Computational Intelligence for Measurement Systems and Applications (CIMSAs 2009), 11-13 May 2009, Piscataway, NJ, USA 2009. 2009 IEEE International Conference on Computational Intelligence for Measurement Systems and Applications (CIMSAs 2009), pp. 91-96. IEEE
84. Justh, E.W., Krishnaprasad, P.S.: Equilibria and steering laws for planar formations. *Systems & Control Letters* **52**(1), 25-38 (2004).
85. Lindhe, M., Ogren, F., Johansson, K.H.: Flocking with obstacle avoidance: A new distributed coordination algorithm based on Voronoi partitions. In: 2005 IEEE International Conference on Robotics and Automation, April 18, 2005 - April 22, 2005, Barcelona, Spain 2005. Proceedings - IEEE International Conference

- on Robotics and Automation, pp. 1785-1790. Institute of Electrical and Electronics Engineers Inc.
86. Gutman, S.: UNCERTAIN DYNAMICAL SYSTEMS - A LYAPUNOV MIN-MAX APPROACH. IEEE Transactions on Automatic Control **AC-24**(Compendex), 437-443 (1979).
 87. Corless, M.J., Leitmann, G.: CONTINUOUS STATE FEEDBACK GUARANTEEING UNIFORM ULTIMATE BOUNDEDNESS FOR UNCERTAIN DYNAMIC SYSTEMS. IEEE Transactions on Automatic Control **AC-26**(Compendex), 1139-1144 (1981).
 88. Ball, J.A., Helton, J.W., Walker, M.L.: H control for nonlinear systems with output feedback. IEEE Transactions on Automatic Control **38**(Compendex), 546-559 (1993).
 89. Sontag, E.D.: Smooth stabilization implies coprime factorization. IEEE Transactions on Automatic Control **34**(Compendex), 435-443 (1989).
 90. Freeman, R.A., Kokotović, P.V.: Robust nonlinear control design : state-space and Lyapunov techniques. Systems & control. Birkhäuser, Boston (1996)
 91. Lyapunov, A.M.: Probleme général de la stabilité du mouvement. Ann. Fac. Sci **9**, 203-474 (1907).
 92. Kanellakopoulos, I., Kokotovic, P.V., Morse, A.S.: Systematic design of adaptive controllers for feedback linearizable systems. IEEE Transactions on Automatic Control **36**(Compendex), 1241-1253 (1991).
 93. Freeman, R.A., Kokotovic, P.V.: Backstepping design of robust controllers for a class of nonlinear systems. In: Proceedings of the IFAC Symposium on Nonlinear Control Systems Design 1992, June 24, 1992 - June 26, 1992, Bordeaux, Fr 1993. IFAC Symposia Series, pp. 431-436. Publ by Pergamon Press Inc
 94. Freeman, R.A., Kokotovic, P.V.: Design of 'softer' robust nonlinear control laws. Automatica **29**(Copyright 1994, IEE), 1425-1437 (1993).
 95. Ogren, P., Egerstedt, M., Hu, X.: A control Lyapunov function approach to multiagent coordination. IEEE Transactions on Robotics and Automation **18**(Compendex), 847-851 (2002).
 96. He, Y., Han, J.: Decentralized receding horizon control for multiple Unmanned Helicopters considering dynamics model. In: 2009 Joint 48th IEEE Conference on Decision and Control (CDC) and 28th Chinese Control Conference (CCC 2009), 15-18 Dec. 2009, Piscataway, NJ, USA 2009. 2009 Joint 48th IEEE Conference on Decision and Control (CDC) and 28th Chinese Control Conference (CCC 2009), pp. 8351-8356. IEEE
 97. Vicsek, T., Czirok, A., Ben-Jacob, E., Cohen, I., Shochet, O.: Novel type of phase transition in a system of self-driven particles. Physical Review Letters **75**(Compendex), 1226-1226 (1995).
 98. Das, A., Subbarao, K., Lewis, F.: Dynamic inversion of quadrotor with zero-dynamics stabilization. In: 2008 IEEE International Conference on Control Applications (CCA) part of the IEEE Multi-Conference on Systems and Control,

- 3-5 Sept. 2008, Piscataway, NJ, USA 2008. 2008 IEEE International Conference on Control Applications (CCA) part of the IEEE Multi-Conference on Systems and Control, pp. 1189-1194. IEEE
99. Jian, W., Thomas, B., Michael, A., Leonhard, H., Florian, H., Weihua, Z., Tiau Hiong, G.: Attitude Free Position Control of a Quadcopter using Dynamic Inversion. Paper presented at the Infotech@Aerospace 2011, St. Louis, Missouri, 29 - 31 March 2011
 100. Lancaster, P., Rodman, L.: Algebraic Riccati equations / Peter Lancaster and Leiba Rodman. Oxford science publications. Oxford : Clarendon Press ; New York : Oxford University Press, 1995., (1995)